Parameterized Reunion with Achromatic Number

Satyabrata Jana
University of Warwick, UK

Souvik Saha
The Institute of Mathematical Sciences, HBNI, Chenai, India

Saket Saurabh
The Institute of Mathematical Sciences, HBNI, Chenai, India
University of Bergen, Norway

Anannya Upasana
The Institute of Mathematical Sciences, HBNI, Chenai, India

Abstract

In this paper, we study the ACHROMATIC NUMBER problem. Given a graph G and an integer k, the task is to determine whether there exists a proper coloring of G, using at least k colors, in which every pair of distinct colors appears on the endpoints of some edge. It was established early on that the problem is fixed-parameter tractable (FPT)– even before the formal development of parameterized complexity. In fact, Farber, Hahn, Hell, and Miller [JCTB, 1986] devised an algorithm with a running time of $\mathcal{O}(f(k) \cdot |E(G)|)$. Although the exact form of $f(k)$ was not specified, it appears to be at least doubly exponential in k. In our work, we first present an algorithm with an explicit dependence on k, and then introduce another algorithm that is parameterized by the vertex cover number of the graph. More formally, we show the following.

- ACHROMATIC NUMBER is solvable in time $2^{\mathcal{O}(k^5)} + \mathcal{O}(|E(G)|)$.
- ACHROMATIC NUMBER admits a polynomial kernel when the input is restricted to a d-degenerate graph and a more efficient kernel on trees.
- We also study the parameterized complexity of the problem with respect to VERTEX COVER and show that it admits an FPT algorithm running in time $2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)}$, where ℓ is the size of a vertex cover.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Achromatic number, Coloring, Fixed-parameter tractability, Kernelization, Lower bound, W-hardness

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2025.42

Funding *Satyabrata Jana*: Supported by the Engineering and Physical Sciences Research Council (EPSRC) via the project MULTIPROCESS (grant no. EP/V044621/1).
Saket Saurabh: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416).

1 Introduction

In this paper, we revisit the ACHROMATIC NUMBER problem that had a fixed parameter tractable (FPT) algorithm even before the concept of parameterized complexity was established, as we currently understand it. A *proper coloring* of a graph is an assignment of colors to the vertices such that no two endpoints of an edge are assigned the same color. A *complete coloring* of a graph is a proper coloring of a graph such that for any pair of distinct colors, there exists a pair of adjacent vertices that have been assigned those colors. The *achromatic number* of a graph G, denoted by $\psi(G)$, is the maximum number of colors that can be used in a complete coloring of G. In this paper, we study the following problem.

Achromatic Number **Parameter:** k
Input: A graph G on n vertices and m edges and an integer k.
Question: Is $\psi(G) \geq k$?

The computational complexity of Achromatic Number was first considered by Yannakakis and Gavril in 1980, who showed that the problem is NP-complete [22, Corollary 2]. The authors also observed that, unlike the classical Chromatic Number problem, which is known to be NP-complete even for $k = 3$ (that is, whether the input graph can be properly colored with 3 colors), Achromatic Number cannot be shown to be NP-complete for a fixed value of k. Toward this, they show that for a graph G, $\psi(G) \geq k$ if and only if there is a subset $W \subseteq V(G)$ of size at most $\binom{k}{2}$ such that $\psi(G[W]) \geq k$. This immediately leads to an algorithm with running time $\mathcal{O}(n^{k^2})$. In the modern terminology of parameterized complexity (PC), this is an XP algorithm for Achromatic Number. Thus, a natural question is whether there exists an algorithm with running time $f(k)n^{\mathcal{O}(1)}$, that is, whether the problem admits an FPT algorithm.

A natural question is whether there exists a uniform polynomial-time algorithm for every fixed k. That is, do we have $n^{\mathcal{O}(1)}$ time algorithm where $\mathcal{O}(1)$ does not depend on k. (Pre PC Era)
In modern terminology, does there exist an algorithm with running time $f(k) \cdot n^{\mathcal{O}(1)}$, that is, does the problem admit an FPT algorithm. (Post PC Era)

In 1986, Farber, Hahn, Hell and Miller [10] designed a "linear time algorithm" for Achromatic Number. That is, an algorithm with running time $\mathcal{O}(m)$; which is actually an algorithm with running time $f(k) \cdot m$. This established the parameterized complexity of Achromatic Number. The exact value of $f(k)$ in the running time is not given and it appears to be at least doubly exponential in k. This algorithm was built on an earlier work of Hell and Miller [14] done in 1976 who showed that the number of irreducible graphs (without any twins) with $\psi(G) = k$ is upper bounded by some function $h(k)$. These results are the starting point of our research. Our aim is to design an algorithm with explicit dependence on k by a function that grows as slowly as possible. Furthermore, we also explore the problem with a structural parameter.

Related Works. Bounds on Achromatic Number in terms of other graph variants like vertex cover number and independence number have been studied [5, 12]. The Pseudo-achromatic Number problem, a related problem, was introduced in 1969 [13] and has been studied extensively [3, 4, 21]. The Pseudo-achromatic Number problem or Graph Complete Partition problem checks whether an undirected graph can be partitioned into k classes such that every pair of classes is connected by an edge. Unlike the achromatic number problem, we do not require these classes to be independent sets. The pseudo-achromatic number may strictly exceed the achromatic Number even for a family of trees [9]. Halldórsson et al studied the problem from the approximation viewpoint, giving tight lower and upper bounds on its approximability. The problem has a $\mathcal{O}(k^2)$ kernel that also establishes that it is FPT parameterized by the solution size [7]. Another related problem is the Harmonious Coloring where the objective is to find a proper coloring of the graph such that each pair of colors appears together on at most one edge [15]. It is known to be NP-hard on trees, interval and permutation graphs [8, 2, 20]. Georges investigated the problem on paths, cycles, complete graphs and complete bipartite graphs [11]. Kolay et al [17] studied it from an

FPT perspective and showed that it is FPT parameterized by the solution size as well as the vertex cover number of the graph. They also gave an exact exponential time algorithm in split graphs.

1.1 Our Results and Methods with Solution Size as a Parameter

Our first main result is the following FPT algorithm for ACHROMATIC NUMBER.

▶ **Theorem 1.** ACHROMATIC NUMBER *can be solved in* $2^{\mathcal{O}(k^5)} + \mathcal{O}(m)$ *time.*

The idea behind the proof of Theorem 1 is motivated by results in the area of approximation algorithms for an optimization variant of ACHROMATIC NUMBER. Kortsarz and Krauthgamer gave an algorithm that approximates the achromatic number within a ratio of $\mathcal{O}\left(\frac{n \cdot \log\log n}{\log n}\right)$ for general graphs [18]. This algorithm in turn was motivated by an earlier combinatorial work of Máté [19] on ACHROMATIC NUMBER on irreducible graphs. Our parameterized algorithm builds on this approximation algorithm and incorporates new ideas and concepts in several places to transform it into an FPT algorithm. We also mention in passing that Kortsarz and Krauthgamer [18] showed that there is no $(2-\epsilon)$-approximation algorithm for every fixed $\epsilon > 0$, unless P=NP.

The next natural question is whether the problem admits a polynomial kernel. That is, in polynomial time, could we replace the given instance by an equivalent instance of size polynomial in k? We do not know the answer to this question, and we leave this as a challenging open problem. However, could we say something when the input instances are restricted to some structured graph classes? In this regard, we first mention that ACHROMATIC NUMBER is known to be NP-complete even when the input graph is a tree [6]. Thus, researchers have considered ACHROMATIC NUMBER on several classes of trees [10] and designed polynomial time algorithms on these graph classes. In this paper, we design polynomial kernels for ACHROMATIC NUMBER on trees and d-degenerate graphs and obtain the following results.

▶ **Theorem 2 (♠).** ACHROMATIC NUMBER *admits a kernel of size* $\mathcal{O}(k^2)$ *on forests.* [1]

▶ **Theorem 3.** ACHROMATIC NUMBER *admits a kernel of size* $\mathcal{O}(k^{24d+2})$ *on* d-degenerate *graphs.*

1.2 Our Results and Methods for Structural Parameterization

We further study the ACHROMATIC NUMBER problem with respect to other parameters. We define the problem ACHROMATIC NUMBER/MOD(π), where π is a graph class, as follows.

> ACHROMATIC NUMBER/MOD(π) **Parameter:** ℓ
> **Input:** An undirected graph G, a modulator of size ℓ to π and a positive integer k.
> **Question:** Is $\psi(G) \geq k$?

Recall that ACHROMATIC NUMBER is known to be NP-complete even when the input graph is a tree [6]. This immediately rules out the treewidth or the size of the feedback vertex set as parameters.

[1] Due to space constraints, the proofs of results marked with ♠ have been omitted and will appear in the full version of the paper.

Modulator to edgeless graphs. Our parameter of study is the vertex cover number of the graph, that is, a vertex set whose deletion results in an edgeless graph. Here, π is the family of edgeless graphs. Our result in this direction is the following.

▶ **Theorem 4.** Achromatic Number/VC *can be solved in* $2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)}$ *time.*

To prove Theorem 4, we first show that the achromatic number of the graph is bounded by $\ell + 1$. Then, we guess the color of the vertices in the modulator (one that will result in a solution). It is possible that there exists a pair of colors that is not assigned to the endpoints of any edge, for e.g., there may not be an edge between color classes C_1 and C_2. To address this issue, we utilize a vertex v outside the modulator and assign it either color C_1 or C_2, thereby *resolving the pair* formed by color classes C_1 and C_2 (in essence, we resolve the conflict between color classes C_1 and C_2 that arises because there is no edge between the color classes). We solve this by reducing the problem to an instance of Subgraph Isomorphism (given a host graph G and a pattern graph H, find a copy of H in G), where the pattern graph H has size $\mathcal{O}(\ell^2)$ and treewidth $\mathcal{O}(1)$. It is known that Subgraph Isomorphism can be solved in time $2^{\mathcal{O}(|H|)} \cdot |V(G)|^{tw+1}$, where tw is the treewidth of the pattern graph, leading to the desired running time of our algorithm.

2 Preliminaries

In this study, we examine undirected graphs. For a given graph G, we represent its set of vertices as $V(G)$ and its set of edges as $E(G)$. When the context allows, we will refer to these simply as V and E. n refers to the number of vertices and m represents the number of edges. For any vertex $u \in V$, the set of vertices adjacent to u in G is represented by $N_G(u)$. In particular, $N_G(u)$ denotes the open neighborhood of u in G. We will abbreviate this to $N(u)$ when the graph is clear from the context. The degree of a vertex u, denoted as $deg(u)$, refers to the number of vertices in its neighborhood, i.e., $deg(u) = |N(u)|$. We call a graph *irreducible* or *twin-free* if any pair of distinct vertices has distinct open neighborhoods. A graph is called d-degenerate if every subgraph has a vertex with degree at most d. An induced matching or an independent matching is a set of edges such that no two endpoints of distinct edges of the matching are adjacent to each other. A *semi-independent matching* M is a set of edges $\{(x_1, y_1), (x_2, y_2), \ldots, (x_{|M|}, y_{|M|})\}$ such that the sets $X = \{x_1, x_2, \ldots, x_{|M|}\}$ and $Y = \{y_1, y_2, \ldots, y_{|M|}\}$ are independent and for any x_i and y_j with $i < j$, x_i is not adjacent to y_j (see Figure 1). We call a matching M maximal if, after adding any edge in $E(G) \backslash M$ to M, M does not remain a matching. An independent set is a set of vertices that are pairwise non-adjacent. An independent set $S \subseteq V(G)$ is called maximal if adding any vertex from $V(G) \backslash S$ to S destroys its property of being an independent set. For a given proper coloring of a graph, a color class refers to a set of vertices that are assigned the same color. A partial complete coloring of G is a complete coloring of a subset of vertices of G. A partial complete k-coloring is a partial complete coloring with k colors. A *greedy independent partition* of a graph G is an ordered partition of the vertex set $V(G)$ into independent sets, constructed through a greedy process, prioritizing the largest set first while ignoring the order of the rest. At each step, a maximal independent set is selected from the remaining vertices and the process continues until all vertices are covered. The size of a *greedy independent partition* is the number of independent sets in the partition.

3 FPT algorithm for Achromatic Number

In this section, we give an FPT algorithm for Achromatic Number parameterized by solution size in general graphs. We start with describing our kernelization procedure that takes an instance (G, k) of Achromatic Number as input and in $\mathcal{O}(|E(G)|)$ time return an equivalent instance (kernel). We prove the following result.

▶ **Theorem 5.** Achromatic Number *admits a kernel of size* $k^{k+2}2^{k^3+k+2}e^{k^2}$*. And moreover we can find it in* $\mathcal{O}(m)$ *time where* m *is the number of edges in the given graph.*

Our kernelization procedure for a given instance (G, k) involves the following steps. Let R be an equivalence relation defined as follows: for a vertex $v \in V(G)$, the class R_v contains the vertices $\{u \mid N_G(u) = N_G(v)\}$.

1. If $n \leq k^{k+2}2^{k^3+k+2}e^{k^2}$, we return G as kernel.
2. Using Lemma 13, we check in $\mathcal{O}(|E(G)|)$ time whether the number of equivalence classes exceeds $k^{k+1}2^{k^3+k+2}e^{k^2}$, or else obtain their count q.
   a. If the number of equivalence classes exceeds $k^{k+1}2^{k^3+k+2}e^{k^2}$, return a trivial YES-instance of Achromatic Number.
   b. Otherwise, $q \leq k^{k+1}2^{k^3+k+2}e^{k^2}$. Apply Reduction Rules 1 and 2 exhaustively, and return the reduced instance as a kernel of size $k^{k+2}2^{k^3+k+2}e^{k^2}$.

We now show the correctness of this procedure. Toward that, we have to prove two things. First, we need to show that if the number of equivalence classes in G is more than $k^{k+1}2^{k^3+k+2}e^{k^2}$ then $\psi(G) \geq k$ (Lemma 15). And secondly, we have to show that after exhaustive application of the Reduction Rules 2 and 1 the size of $V(G)$ gets reduced to $k^{k+2}2^{k^3+k+2}e^{k^2}$ (Lemma 16). An example of a trivial YES instance of Achromatic Number is $(K_{1,1}, 1)$. We now describe the Reduction Rule 1 that removes all isolated vertices from the graph. The proof of the rule follows from the fact that removing isolated vertices from the graph will not affect its achromatic number, since no edge in the graph is incident to that vertex.

▶ **Reduction Rule 1.** *If G has an isolated vertex v, then return the instance $(G - v, k)$.*

Now mention two known facts regarding complete coloring of a graph.

▶ **Lemma 6** ([18, Lemma 1.1])**.** *A partial complete coloring in a graph G can be extended to a complete coloring of G in time $\mathcal{O}(|E(G)|)$.*

▶ **Lemma 7** ([18, Lemma 1.4])**.** *Given a semi-independent matching of a graph G of size at least $\binom{k}{2}$, a partial complete k-coloring can be computed in time $\mathcal{O}(|E(G)|)$.*

Lemma 6 and Lemma 7 together conclude the following lemma.

▶ **Lemma 8.** *If a graph G has a semi-independent matching of size $\binom{k}{2}$, then $\psi(G) \geq k$.*

We now prove the following crucial lemma.

▶ **Lemma 9.** *Let G be an irreducible graph with $|V(G)| > k^{k+1}2^{k^3+k+2}e^{k^2}$. Then $\psi(G) \geq k$.*

Proof. Let G be an irreducible graph. We show that if G has more than $k^{k+1}2^{k^3+k+2}e^{k^2}$ vertices then $\psi(G) \geq k$. Moreover, we design an algorithm for such a graph, i.e., we give a complete coloring with at least k colors. We first describe the algorithm briefly.

We start with finding a greedy independent partition Π. If the size of Π is at least k, we have that $\psi(G) \geq k$ (by Lemma 17). Moreover Π gives a complete coloring with at least k colors by assigning a distinct color to each set. Next, we focus on the first independent set S_0 in Π. At each recursive step i, we construct $S_{i+1} \subset S_i$ as follows:

- We fix an arbitrary vertex ordering.
- Let x_{2i} and x_{2i+1} be the two minimal vertices with respect to the ordering.
- We identify a *distinguishing vertex* d_{xy} that is adjacent to exactly one of x_{2i} and x_{2i+1}.
- Then we partition S_i into S_i^0 ($\subseteq N(d_{xy})$) and S_i^1 (not intersect $N(d_{xy})$).
- We set $S_{i+1} = S_i^0$ if $|S_i^0| \geq |S_i|/k$; otherwise, set $S_{i+1} = S_i^1$.

This process yields nested independent sets $S_0 \supset S_1 \supset \cdots \supset S_l$, forming either a *partial complete coloring* or a *semi-independent matching*, both can be extendable to a complete coloring. We also store a pair of sets U^0 and U^1 of witness vertices. Simultaneously we keep two index set Q^0 and Q^1. Next we show that for $|V(G)| \geq k^{k+1}2^{k^3+k+2}e^{k^2}$ the Algorithm 1 always terminates. In addition we show that either of the cases $Q^0 \geq k$ or $Q^1 \geq k^3$ leads to a complete coloring with at least k colors (Claims 11 and 12). A complete description of this above mentioned procedure is given in Algorithm 1. Now we show that Algorithm 1 always terminates. Assume that $p < k$ inside Algorithm 1 as otherwise it returns $\psi(G) \geq k$ so terminates. Note that if the Algorithm 1 does not terminate at Step 3 then the while loop (Step 11) runs at most $(k + k^3)$ steps as per our construction of the set Q_0 and Q_1. Now we show that either $|Q^0| \geq k$ or $|Q^1| \geq k^3$ in the output of Algorithm 1.

$\triangleright$ Claim 10. Let G be an irreducible graph with more than $k^{k+1}2^{k^3+k+2}e^{k^2}$ vertices. If the Algorithm 1 outputs the pair of sets Q_0 and Q_1 then either $|Q^0| \geq k$ or $|Q^1| \geq k^3$.

Proof. We prove this by contradiction. Assume G is an irreducible graph with $n > k^{k+1}2^{k^3+k+2}e^{k^2}$ vertices and the Algorithm 1 outputs the pair of sets Q_0 and Q_1 with $|Q^0| < k$ or $|Q^1| < k^3$. Let $S_{\alpha,\beta}$ denote the independent set obtained after applying Step 14 α times and Step 15 β times, starting from S_0. We claim that

$$|S_{\alpha,\beta}| \geq \left(\frac{1}{2k}\right)^{\alpha} \left(\frac{1}{2}\left(1 - \frac{1}{k}\right)\right)^{\beta} |S_0| \tag{1}$$

Base case: When $\alpha + \beta = 0$, we have $S_{\alpha,\beta} = S_0$, so the inequality trivially holds.

Inductive step: By the construction of the algorithm:

$$|S_{\alpha,\beta}| \geq \min\left\{\frac{1}{2k}|S_{\alpha-1,\beta}|, \frac{1}{2}\left(1 - \frac{1}{k}\right)|S_{\alpha,\beta-1}|\right\}$$

By induction hypothesis:

$$|S_{\alpha-1,\beta}| \geq \left(\frac{1}{2k}\right)^{\alpha-1} \left(\frac{1}{2}\left(1 - \frac{1}{k}\right)\right)^{\beta} |S_0|, \quad |S_{\alpha,\beta-1}| \geq \left(\frac{1}{2k}\right)^{\alpha} \left(\frac{1}{2}\left(1 - \frac{1}{k}\right)\right)^{\beta-1} |S_0|$$

Thus, in either case, i.e., $|S_{\alpha,\beta}| \geq \frac{1}{2k}|S_{\alpha-1,\beta}|$ or $|S_{\alpha,\beta}| \geq \frac{1}{2}\left(1 - \frac{1}{k}\right)|S_{\alpha,\beta-1}|$ Equation (1) is satisfied. Now, since $|Q^0| < k$ and $|Q^1| < k^3$, the loop terminates after at most $k + k^3$ steps. Therefore, the set S_{k,k^3} has size at most 4, but also satisfies:

$$|S_{k,k^3}| \geq \left(\frac{1}{2k}\right)^{k} \left(\frac{1}{2}\left(1 - \frac{1}{k}\right)\right)^{k^3} |S_0|$$

Algorithm 1 Algorithm for the case when the graph is large and irreducible.

Input: An irreducible graph G with $n > k^{k+1}2^{k^3+k+2}e^{k^2}$ vertices.
Output: Either $\psi(G) \geq k$ or a pair of sets Q^0 and Q^1.

1: Compute a greedy independent partition $I_1, \ldots, I_p$ of G
2: **if** $p \geq k$ **then**
3:     **return** $\psi(G) \geq k$
4: Let $I = I_\alpha$ where $|I_\alpha| \geq |I_j|, \forall j \in [p]$, with a total order $\prec$ on I $\triangleright$ $|I| \geq n/k$
5: **for all** $x, y \in I$ **do**
6:     Choose d_{xy} adjacent to exactly one of x, y
7:     $c(x, y) \leftarrow$ color of d_{xy}
8: **for all** $z \succ y \succ x$ in I **do**
9:     $g(x, y, z) \leftarrow \begin{cases} 0 & \text{if } (z, d_{xy}) \in E \\ 1 & \text{otherwise} \end{cases}$
10: Initialize $S_0 = I$, $Q^0 = Q^1 = \emptyset$, $i = 0$
11: **while** $|S_i| > 4$ **do**
12:     $x_{2i} \leftarrow \min_\prec S_i$, $x_{2i+1} \leftarrow$ next in S_i
13:     Partition $S_i \setminus \{x_{2i}, x_{2i+1}\}$ into S_i^0 and S_i^1:

$$S_i^0 = \{x \mid g(x_{2i}, x_{2i+1}, x) = 0\}, S_i^1 = \{x \mid g(x_{2i}, x_{2i+1}, x) = 1\}$$

14:     **if** $|S_i^0| \geq \frac{|S_i|}{2k}$ **then** $S_{i+1} \leftarrow S_i^0$, $Q^0 \leftarrow Q^0 \cup \{i\}$
15:     **else** $|S_i^1| \geq \frac{1}{2}\left(1 - \frac{1}{k}\right)|S_i|$ **then** $S_{i+1} \leftarrow S_i^1$, $Q^1 \leftarrow Q^1 \cup \{i\}$
16:     **if** $|Q^0| \geq k$ or $|Q^1| \geq k^3$ **then**
17:         **break**
18:     $i \leftarrow i + 1$
19: **return** Q^0 and Q^1

Since $p < k$, we have $|S_0| \geq \frac{n}{k}$, giving $|S_{k,k^3}| \geq \left(\frac{1}{2k}\right)^k \left(\frac{1}{2}\left(1 - \frac{1}{k}\right)\right)^{k^3} \frac{n}{k}$. Using the standard inequality $(1 - \frac{1}{k})^{k^3} \geq e^{-k^2}$ for large k, and the condition $|S_{k,k^3}| \leq 4$, we have $n \leq k^{k+1} \cdot 2^{k^3+k+2} \cdot e^{k^2}$. This contradicts our assumption that $n > k^{k+1} \cdot 2^{k^3+k+2} \cdot e^{k^2}$. Therefore, either $|Q^0| \geq k$ or $|Q^1| \geq k^3$ must hold. $\triangleleft$

We now compute a pair of sets U^0 and U^1 as follows: $U^0 = \{(x_{2i}, x_{2i+1}) \mid i \in Q^0\}$ and $U^1 = \{(x_{2i}, x_{2i+1}) \mid i \in Q^1\}$, where Q^0 and Q^1 be the set returned by Algorithm 1.

$\triangleright$ Claim 11. If $|Q^0| \geq k$ then $\psi(G) \geq k$.

Proof. Since $|Q^0| \geq k$, there are at least k tuples of the form $(x_{2i}, x_{2i+1}, d_{x_{2i},x_{2i+1}})$ for $i \in [|Q^0|]$. For each such tuple, let x_h denote the vertex between x_{2i} and x_{2i+1} that is not adjacent to $d_{x_{2i},x_{2i+1}}$. We assign a new color c_i to the pair $(x_h, d_{x_{2i},x_{2i+1}})$, thereby assigning at least k new colors. We now show that this coloring forms a partial $|Q^0|$-complete coloring. From the construction of Q^0 in Step 14 of the algorithm, we know that each vertex $d_{x_{2i},x_{2i+1}}$, for $i \in [|Q^0|]$, is adjacent to both x_{2j} and x_{2j+1} for all $j > i$. Thus, for any pair of colors c_i and c_j with $i > j$, the vertex $d_{x_{2j},x_{2j+1}}$ (colored with c_j) is adjacent to both x_{2i} and x_{2i+1}, at least one of which is colored with c_i. Therefore, there is an edge between every pair of color classes assigned in Step 7. By applying Lemma 6, this partial $|Q^0|$-complete coloring can be extended to a complete coloring of size at least $|Q^0|$. $\triangleleft$

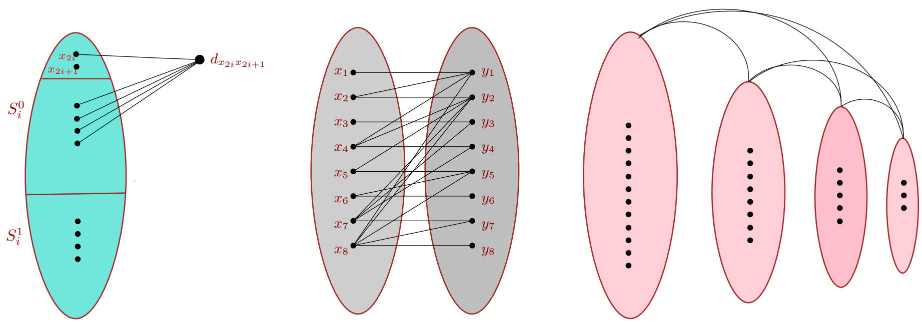

Figure 1 Figure depicting a partition of S_i in Algorithm 1, an instance of a semi-independent matching and a greedy independent partition respectively.

▷ Claim 12. If $|Q^1| \geq k^3$ then $\psi(G) \geq k$.

Proof. Let c' be the color that appears most frequently among the colors assigned to the vertices $d_{x_{2i},x_{2i+1}}$ for all $(x_{2i}, x_{2i+1}) \in U^1$. Define the subset $U' \subseteq U^1$ as the set of all pairs $(x_{2i}, x_{2i+1}) \in U^1$ such that $c(d_{x_{2i},x_{2i+1}}) = c'$. Since the graph G is colored using at most k colors, it follows by the pigeonhole principle that $|U'| \geq k^2$. We now construct a semi-independent matching from U'. For each pair $(x_{2i}, x_{2i+1}) \in U'$, include the vertex $d_{x_{2i},x_{2i+1}}$ and the one among x_{2i} and x_{2i+1} to which it is adjacent. Note that, by the definition of Q^1, any vertex x_j in U^1 with $j > 2i + 1$ is not adjacent to $d_{x_{2i},x_{2i+1}}$. Therefore, the matching constructed in this way is semi-independent and has size at least $k^2 \geq \binom{k}{2}$. Hence, if $|Q^1| \geq k^3$, then we can compute a semi-independent matching of size at least $\binom{k}{2}$. By applying Lemma 8, we conclude that $\psi(G) \geq k$. ◁

The correctness of Step 3 follows from Lemma 17. In Claim 10, we show that if the Algorithm 1 outputs the pair of sets Q_0 and Q_1 then either $|Q^0| \geq k$ or $|Q^1| \geq k^3$. In both the cases, we have that $\psi(G) \geq k$ (by Claim 11 and 12). Hence we are done with showing that if the given graph G is irreducible and more than $k^{k+1}2^{k^3+k+2}e^{k^2}$ vertices then $\psi(G) \geq k$. This completes the proof. ◀

Recall that R was an equivalence relation defined as follows: for a vertex $v \in V(G)$, the class R_v contains the vertices $\{u \mid N_G(u) = N_G(v)\}$.

▶ **Lemma 13** ([10, Theorem 3.3]). *Given a graph G and an integer k, in time $\mathcal{O}(|E(G)|)$, we can*

- *determine if G has at least $f(k)$ equivalence classes, or*
- *build all the equivalence classes.*

Now we define a rule that bounds the number of vertices to each equivalence class.

▶ **Reduction Rule 2.** *If $|R_v| \geq k + 1$, then delete v. Return the instance $(G - v, k)$.*

The correctness of the Reduction Rule 2 follows from the following claim.

▷ Claim 14. Reduction Rule 2 is safe.

Proof. In the forward direction, assume that (G, k) is a YES-instance of Achromatic Number. Then, $\psi(G) \geq k$. If $\psi(G) \geq k + 1$ then $\psi(G - v) \geq k$ trivially holds, as deleting any vertex can decrease the achromatic number by at most one.

So we can assume that $\psi(G) = k$. Let R_v be an equivalence class with at least $k + 1$ vertices $u_1, u_2, \ldots, u_{k+1}$. Without loss of generality, assume that $v = u_{k+1}$. By the definition of an equivalence class, we have $\forall i, j \in [k + 1]$, $N_G(u_i) = N_G(u_j)$. On the contrary, assume that $\psi(G-v) < k$. Let $C_1, \dots, C_k$ be the color classes defined by the complete coloring of G. More precisely, for each $i \in [k]$, the set C_i is the set of all vertices in G with color i. Additionally, assume that $v \in C_k$. Now, if for all $i \in [k-1]$, we have an edge between a pair of vertices in C_i and C_k, excluding v, then we are done. Else, there exists a color class C_i such that, on removal of $v = u_{k+1}$, there is no edge between any pair of vertices in C_i and C_k. Assume that u is a vertex in C_i such that $(u, v) \in E(G)$. Note that in this case the k vertices $u_1, \dots u_k$ are present in the remaining color classes $C_1, \dots C_{k-1}$.

Thus by the pigeon hole principle, there exist u_ℓ and $u_{\ell'}$ in R_v, with $\ell \neq \ell' \in [k]$, which belong to the same color class, say $C_j, j \in [k-1]$. Because $N_G(u_\ell) = N_G(u_{\ell'})$, removing u_ℓ from C_j will not affect the achromatic number of G as $u_{\ell'}$ serves the same purpose as u_ℓ. Now, we can add u_ℓ to C_t. Since u_ℓ and $v = u_{k+1}$ belong to the same equivalence class, they are not connected by an edge and therefore u_ℓ, along with the vertices of C_t forms an independent set. The edge (u, u_ℓ) ensures an edge between the color classes C_i and C_t, and hence the achromatic number of the graph $G - v$ is k. Thus, (G', k) is also a `YES`-instance.

In the backward direction, let (G', k) be a `YES`-instance of Achromatic Number. Then, $\psi(G') \geq k$. Let $C_1, \dots, C_{k'}$, where $k' \geq k$ be the partition of color classes corresponding to a complete coloring of G'. If for some $i \in [k']$, v does not have a neighbor in the vertices corresponding to C_i, then we can assign the color i to v and get a complete coloring of G with at least k colors. Otherwise, for each $i \in [k']$, the color class C_i contains a vertex from $N_G(v)$. Then, we assign a new color $k'+1$ to v and obtain a complete coloring of G with $k'+1$ colors. Hence, in both cases, we obtain a complete coloring on G with at least k colors. Thus, (G, k) is a `YES`-instance. ◁

▶ **Lemma 15.** *If the number of equivalence classes in G is strictly greater than $k^{k+1}2^{k^3+k+2}e^{k^2}$, then $\psi(G) \geq k$.*

Proof. Let $V(G)$ be partitioned into q equivalence classes. Assume $q > k^{k+1}2^{k^3+k+2}e^{k^2}$. Construct a subgraph $G[S]$, where $S \subseteq V(G)$ contains exactly one arbitrarily chosen vertex from each equivalence class. Clearly, $|S| = q$. Now we show that no two distinct vertices in S have the same open neighborhood. Suppose, for contradiction, that there exist distinct vertices $u, v \in S$ such that $N(u) = N(v)$. Then, by the definition of the equivalence relation, u and v must belong to the same equivalence class. However, since S contains exactly one vertex from each equivalence class, this contradicts the assumption that both u and v are in S. Therefore, all vertices in S have distinct open neighborhoods. Therefore, $G[S]$ is an irreducible graph. Since $q > k^{k+1}2^{k^3+k+2}e^{k^2}$, it follows that $|S| > k^{k+1}2^{k^3+k+2}e^{k^2}$. By applying Lemma 9, we obtain $\psi(G[S]) \geq k$. This gives a partial complete coloring of G with at least k colors. Finally, by applying Lemma 6, we conclude that $\psi(G) \geq k$. ◀

Let (G, k) be an instance where none of Reduction Rules 1 and 2 is applicable. Then each equivalence class is bounded by k. This imply the following lemma.

▶ **Lemma 16.** *Let (G, k) be an instance where none of Reduction Rules 1 and 2 is applicable. If the number of equivalence classes in G is at most $k^{k+1}2^{k^3+k+2}e^{k^2}$, then $|V(G)| \leq k^{k+2}2^{k^3+k+2}e^{k^2}$.*

▶ **Lemma 17.** *For any graph G, if size of greedy independent partition is at least k, then $\psi(G) \geq k$.*

Proof. We construct a *greedy independent partition* of G, say $I_1, I_2, \dots, I_p$. We color each independent set with a different color. Since the size of the *greedy independent partition* is at least k, we use at least k colors. Now, from the construction of the *greedy independent* *partition*, observe that all vertices in G outside I_1 must be adjacent to at least one vertex in I_1. Similarly, for each $i > 1$, the neighborhood of all vertices in I_i includes all vertices in $\bigcup_{j=i+1}^{p} I_j$. Thus, there exists no pair of independent sets in a *greedy independent partition* of G such that their union is also an independent set. Hence, our coloring is a complete coloring of size at least k, that is, $\psi(G) \geq k$. ◀

FPT algorithm for Achromatic Number**.** We first compute a kernel of size at most $k^{k+2}2^{k^3+k+2}e^{k^2}$ in $\mathcal{O}(m)$ time (by Theorem 5). It is easy to observe that if a graph admits a complete coloring with at least k colors, then there exists an induced subgraph $H \subseteq G$ with at most $\binom{k}{2}$ vertices that also admits a complete k-coloring. Therefore, on the kernelized instance, we can perform a brute-force search: enumerate all subsets of at most $\binom{k}{2}$ vertices and check whether any of them admits a complete coloring with at least k colors. Each such check can be performed in $\mathcal{O}(k^2)$ time. If such a subgraph is found, we return `YES` for the original instance (by Lemma 6); otherwise, we return `NO`. The kernelization step takes $\mathcal{O}(m)$ time, and the brute-force step takes at most $\binom{k^{k+2}2^{k^3+k+2}e^{k^2}}{\binom{k}{2}} \cdot \mathcal{O}(k^2) = 2^{\mathcal{O}(k^5)}$ time. Hence, the total running time of the algorithm for solving Achromatic Number is $2^{\mathcal{O}(k^5)} + \mathcal{O}(m)$.

▶ **Theorem 1.** Achromatic Number *can be solved in* $2^{\mathcal{O}(k^5)} + \mathcal{O}(m)$ *time.*

4 Parameterized by Vertex Cover

In this section, we study the parameterized complexity of Achromatic Number with respect to a structural parameter that is a modulator to edgeless graphs (also known as vertex cover). We call this version of the problem Achromatic Number/VC which is formally defined below.

> Achromatic Number/VC **Parameter:** ℓ
> **Input:** An undirected graph G, a set $S \subseteq V(G)$ of size ℓ such that $G - S$ is an independent set and a positive integer k.
> **Question:** Is $\psi(G) \geq k$?

The following is the main result of this section.

▶ **Theorem 4.** Achromatic Number/VC *can be solved in* $2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)}$ *time.*

▶ **Observation 18.** *If G has a vertex cover of size at most ℓ, then $\psi(G) \leq (\ell + 1)$.*

Proof. Let S be a vertex cover of size at most ℓ in G. Suppose, for a contradiction, $\psi(G) > (\ell + 1)$. Then, there exists a complete coloring $\mathcal{C}$ using at least $\ell + 2$ colors. Since $|S| \leq \ell$, at most ℓ color classes of $\mathcal{C}$ contain the vertices of the set S. This implies that there are at least two color classes, say C_i and C_j, which do not contain any vertices of S. Now the subgraph of G induced by the vertices of $C_i \cup C_j$ is an independent set, which implies that there is no edge across the color classes C_i and C_j. This contradicts the fact that $\mathcal{C}$ is a complete coloring. ◀

An input instance of our problem is (G, S, k, ℓ), where S is a vertex cover of size at most ℓ. Furthermore, assume that (G, S, k, ℓ) is a `YES` instance and $\Pi = (X_1, X_2, \ldots, X_k)$ is a hypothetical solution where Π is a partition of vertices into k independent sets $X_1, X_2, \ldots, X_k$ and, for each pair i, j there is a pair of vertices $u \in X_i$ and $v \in X_j$ such that $(u, v) \in E(G)$.

Our ultimate objective is to get Π. To do this, we try to obtain as much information as possible about Π in time $\ell^\ell \cdot n^{\mathcal{O}(1)}$. Observe that if S is an empty set, then (G, S, k, ℓ) is a `YES` instance if and only if $k = 1$, and we can obtain Π by simply setting $X_1 = V(G)$. In what follows, we assume that we are given an input (G, S, k, ℓ) and a partition $\pi = (Y_1, Y_2, \dots, Y_k)$ of S, and our problem is to test whether π can be *extended* to the desired partition Π. More specifically, we test whether there is a feasible solution, that is, partition $\Pi = (X_1, X_2, \dots, X_k)$ of $V(G)$ such that $Y_i \subseteq X_i$, for each $1 \le i \le k$. This leads us to the following problem.

> Disjoint Achromatic Number/vc **Parameter:** ℓ
> **Input:** A graph G, a set $S \subseteq V(G)$ of size at most ℓ such that $G - S$ is an independent set, an integer k, and a partition $\pi = (Y_1, Y_2, \dots, Y_k)$ of S.
> **Question:** Does there exist a solution $\Pi = (X_1, X_2, \dots, X_k)$ with the requisite properties that extends π?

We use (G, S, k, ℓ, π) to denote an instance of Disjoint Achromatic Number/vc.

Our next lemma formally proves our discussion by showing that Achromatic Number/VC and Disjoint Achromatic Number/vc are FPT equivalent. That is, Achromatic Number/VC is FPT if and only if Disjoint Achromatic Number/vc is FPT.

▶ **Lemma 19.** *Let G be a graph and S be a vertex cover of size at most ℓ. For any integer k, (G, S, k, ℓ) is a* `YES`*-instance of* Achromatic Number/VC *if and only if either (G, S, k, ℓ, π) is a* `YES`*-instance or there exists a non-empty set $X \subseteq V(G) \setminus S$ such that $|X| \le \ell$ and $(G, S \cup X, k, 2\ell, \pi)$ is a* `YES`*-instance of* Disjoint Achromatic Number/vc*, for some partition π of S or $S \cup X$ respectively.*

Proof. The backward direction is obvious due to the definition of Disjoint Achromatic Number/vc. If either (G, S, k, ℓ, π) or $(G, S \cup X, k, 2\ell, \pi)$ for some $X \subseteq V(G) \setminus S$ is `YES`, then by definition we have a feasible solution with at least k colors that extends π.

In the forward direction, let (G, S, k, ℓ) be a `YES` instance of Achromatic Number/VC, then there exists a partition of G, say $C_1, \dots, C_t$, with $t \ge k$, such that between any two pairs of color classes C_i and C_j, $i \ne j \in [t]$, there exist vertices $u \in C_i$ and $v \in C_j$ with $(u, v) \in E(G)$. Observe that there can be at most one color class that does not contain any vertices from S. Now there are two cases.

Case 1. Every color class C_i intersects S. In this case, we define a partition $\pi = (Y_1, Y_2, \dots, Y_t)$ of S as follows. For each $i \in [t]$ the set $Y_i \coloneqq S \cap C_i$. Hence (G, S, k, ℓ, π) is a `YES` instance.

Case 2. There is a color class that does not intersect S. Wlog, assume that C_1 is such a color class. Note $t \le \ell + 1$. Observe that for every set C_i, $2 \le i \le t$, there is a vertex in C_1 that has a neighbor in C_i. There may be many vertices that has a neighbor in C_i but we choose an arbitrary vertex $u_i \in C_1$ that has a neighbor in C_i. In this process, for each i, $2 \le i < [t]$, we find a vertex $u_i \in C_1$. Let $X = \{u_2, \dots, u_q\}$, where $q \le t$. As $q \le t \le \ell + 1$, so $|X| \le \ell$. Clearly, $S \cup X$ is a vertex cover of size at most 2ℓ. Now we define a partition $\pi = (Y_1, Y_2, \dots, Y_k)$ of $S \cup X$ as follows. For $i = 1$, define $Y_1 \coloneqq X$, for all other i, $2 \le i \le [t]$ we define the set $Y_i \coloneqq S \cap C_i$. Hence $(G, S \cup X, k, 2\ell, \pi)$ is a `YES` instance. ◀

Next, we aim to generate a collection $\mathcal{I}$ of $f(\ell)$ many instances for Disjoint Achromatic Number/vc from an input instance (G, S, k, ℓ) of Achromatic Number/VC. First, for each partition π of S, we include an instance (G, S, k, ℓ, π) into $\mathcal{I}$. Next, consider the equivalence relation R defined for Lemma 13. Given that $|S| \le \ell$, it follows that the number of equivalence classes is at most 2^ℓ. Thus, for any pair of vertices u, v in the same equivalence class, we have $N(u) = N(v)$. Let Z^* be a set of vertices formed by arbitrarily choosing a vertex from each equivalence class. Since the number of equivalence classes is at most 2^ℓ, it follows that $|Z^*| \leq 2^\ell$. Now for each subset $Z \subseteq Z^*$ where $|Z| \leq \ell$, and for each partition π of $S \cup Z$, we add an instance $(G, S \cup Z, k, 2\ell, \pi)$ to $\mathcal{I}$. Therefore, the total number of instances is bounded by $\ell^\ell + \binom{2^\ell}{\ell} \cdot (\ell+1)^{2\ell} \leq 2^{\mathcal{O}(\ell^2)}$. The following claim is derived from the arguments of the proof of Lemma 19 which basically tells that these two problems are FPT equivalent.

▷ Claim 20. (G, S, k, ℓ) is YES instance if and only if one of the instances of $\mathcal{I}$ is YES.

4.1 Algorithm for Disjoint Achromatic Number/vc

Let (G, S, k, ℓ, π) be an instance of Disjoint Achromatic Number/vc, where $\pi = (Y_1, Y_2, \ldots, Y_k)$. Our algorithm works as follows. First, the algorithm performs a simple sanity check reduction rule. In essence, it checks whether π is valid.

▶ **Reduction Rule 3.** *Return that (G, S, k, ℓ, π) is a* NO *instance of* Disjoint Achromatic Number/vc*, if one of the following holds:*
1. *$S = \emptyset$ and $k \geq 2$.*
2. *There is a set Y_i in $\pi = (Y_1, \ldots, Y_k)$ such that Y_i is not an independent set.*
3. *There is a pair of sets Y_i and Y_j in $\pi = (Y_1, \ldots, Y_k)$ such that $Y_i \cup Y_j \cup (V - S)$ is an independent set.*

▶ **Lemma 21.** *Reduction Rule 3 is safe.*

Proof.
1. If $S = \emptyset$, then G is an independent set. If there are at least 2 color classes, then there is no edge going across any pair of color classes. Therefore, (G, S, k, ℓ, π) is a NO instance of Disjoint Achromatic Number/vc for $k \geq 2$.
2. If there is a set Y_i in $\pi = (Y_1, \ldots, Y_k)$ such that Y_i is not an independent set, then if we extend π to Π, a partition of G, we get $X_i \supseteq Y_i$ which is not an independent set. This violates the property of complete coloring.
3. If there are two sets Y_i and Y_j in π, then an edge that goes across the pair of color classes Y_i and Y_j must have endpoints in Y_i and Y_j, Y_i and $V - S$ or Y_j and $V - S$. However, the graph induced on $Y_i \cup Y_j \cup (V - S)$ is an independent set. Therefore, (G, S, k, ℓ, π) is a NO instance of Disjoint Achromatic Number/vc. ◀

Now we describe our algorithm. We first find the bad pairs in $[k] \times [k]$. We say that a pair (i, j) is a bad pair if and only if $Y_i \cup Y_j$ induces an independent set in G. It is easy to observe that the number of bad pairs is at most k^2.

We now construct two graphs; one is the *host graph*, denoted by G_1, and another is the *pattern graph* G_2 satisfying the condition as follows: G_1 has a subgraph isomorphic to G_2 if and only if (G, S, k, ℓ, π) is an YES-instance. Let b be the number of bad pairs in $[k] \times [k]$ and $t = |V \setminus S|$. See Figure 2 and Figure 3 for an illustration of the construction.

Construction of host graph G_1

1. We add two sets $A_1 = \{w_1, \ldots, w_b\}$ and $B_1 = \{u_1, \ldots, u_b\}$, each of the b vertices in $V(G_1)$ corresponding to each bad pair in $[k] \times [k]$.
2. We add a set $I_1 = \{v_1, \ldots, v_t\}$ of t vertices to $V(G_1)$, one for each vertex in $V \setminus S$.

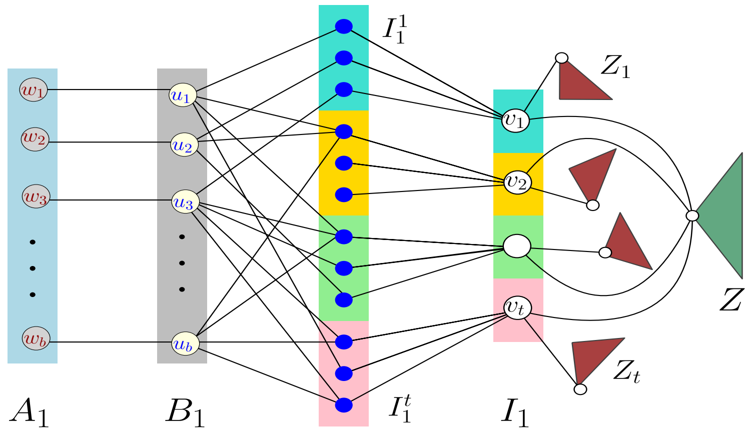

Figure 2 An illustration of the construction of host graph.

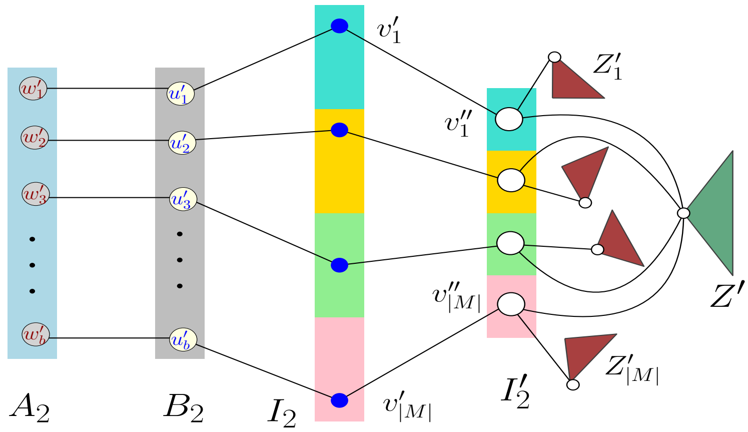

Figure 3 An illustration of the construction of pattern graph.

3. We add a set $I_1^i = \{v_i^1, \ldots, v_i^k\}$ of k vertices for each $1 \leq i \leq t$ in $V(G_1)$.
4. We add t cliques $Z_1, \ldots, Z_t$, each of size 50 in G_1. For each clique Z_i take an arbitrary vertex in it and make it adjacent to $v_i \in I_1$.
5. We add a clique Z of size 100 to G_1. Choose an arbitrary vertex in Z and make it adjacent to each vertex in I_1.
6. Add the set $\{(u_i, w_i) \colon i \in [b]\}$ of edges to $E(G_1)$.
7. Add the set $\{(v_i, v_i^j) \colon i \in [t], j \in [k]\}$ of edges to $E(G_1)$.
8. For any pair of integers i and j, we add an edge between $v_i^j \in I_1^i$ and a vertex $u \in B_1$, where u is a bad pair corresponding to (c, j) or (j, c) if (i) v_i has no neighbor in Y_j and (ii) v_i has a neighbor in Y_c.

Observe here that in step 8 when we add an edge between $v_i^j \in I_1^i$ and a vertex $u \in B_1$, if the vertex v_i is assigned a color j, that is, v_i is added to the partition Y_j then it satisfies the bad pair corresponding to u. Here, we say that a vertex satisfies a bad pair when its addition to one of the partitions of the pair introduces an edge between the pair. Our goal is to check if there exists an assignment of colors to the vertices in $G \setminus S$ that satisfies all the bad pairs.

We create 2^b many pattern graphs as follows: Let M be a minimal set of vertices in $\bigcup_{i=1}^{t} I_1^i$ such that (i) $M \cap I_1^i \leq 1$ for each $1 \leq i \leq t$, (ii) $N(u) \cap M \neq \emptyset$ for any $u \in B_1$. Clearly $|M| \leq b$. We guess the value of $|M|$. Then we guess $|M|$ numbers such that their sum is b. We use the following fact to bound the number of such partitions.

▶ **Fact 1.** *For any positive integers n and k the number of k tuples of positive integers whose sum is n equals $\binom{n-1}{k-1} \leq 2^n$.*

Let us guess the value of $|M|$ and $|M|$ numbers $n_1, n_2, \ldots, n_{|M|}$ whose sum equals b. Each value n_i is essentially the number of bad pairs that a vertex will satisfy privately. We will later describe the notion of satisfying privately. In the following, we describe how to construct a pattern graph for such a guess. Putting $n = b$ in Fact 1, we can decide that the number of such partitions is at most 2^b. Let $\mathcal{H}$ denote the set of all pattern graphs.

Construction of pattern graph $G_2 \in \mathcal{H}$: For $n_1 + n_2 + \ldots + n_{|M|} = b$

1. We add two sets $A_2 = \{w'_1, \ldots, w'_b\}$ and $B_2 = \{u'_1, \ldots, u'_b\}$, each of the b vertices in $V(G_2)$.
2. We add a set $I_2 = \{v'_1, v'_2, \ldots, v'_{|M|}\}$ of $|M|$ vertices to $V(G_2)$.
3. We add a set $I'_2 = \{v''_1, v''_2, \ldots, v''_{|M|}\}$ of $|M|$ vertices to $V(G_2)$.
4. We add $|M|$ cliques $Z'_1, \ldots, Z'_{|M|}$, each of size 50 in G_2. For each clique Z'_i take an arbitrary vertex in it and make it adjacent to $v''_i \in I'_2$.
5. We add a clique Z' of size 100 in G_2. Choose an arbitrary vertex in Z' and make it adjacent to each vertex in I'_2.
6. Add the set $\{(u'_i, w'_i) : i \in [b]\} \bigcup \{(v'_i, v''_i) : i \in [|M|]\}$ of edges to $E(G_2)$.
7. Add the set $\{(v'_i, u'_j) : i \in [|M|], (n_1 + \ldots + n_{i-1}) + 1 \leq j \leq (n_1 + \ldots + n_i), n_0 = 0\}$ of edges to $E(G_2)$.

▷ Claim 22. $\mathsf{tw}(G_2) = \mathcal{O}(1)$.

Proof. Consider the graph $G_2^{'}$ without copies of cliques K_{100} and K_{50}. Let (T, β) be the tree decomposition of $G_2^{'}$. The treewidth of $G_2^{'}$ is 1 as it is a tree. We add the cliques K_{100} and K_{50} to every bag of T. This gives us a tree decomposition of G_2 with a constant treewidth. ◁

▶ **Observation 23.** $|V(G_2)| = \mathcal{O}(k^2)$.

Proof. We know that $|A_2| = |B_2| \leq b \leq k^2$, and $|I_2| = |I'_2| \leq b \leq k^2$. There are $|M|$ cliques of size 50 each and a clique of size 100. Thus, $|V(G_2)| \leq 4k^2 + 50k^2 + 100 = \mathcal{O}(k^2)$. ◀

The following lemma proves the correctness of our algorithm.

▶ **Lemma 24.** *(G, S, k, ℓ, π) is a* YES *instance of* Disjoint Achromatic Number/vc *if and only if there exists a subgraph of G_1 which is isomorphic to G_2 for some $G_2 \in \mathcal{H}$.*

Proof. Assume (G, S, k, ℓ, π) is a YES instance of Disjoint Achromatic Number/vc. Then there is a partition π of S that can be extended to a partition Π of G (each part of a partition corresponds to a color class) such that there is an edge across any pair of color classes in the partition Π. We show the existence of a pattern graph G_2 that is isomorphic to a subgraph of G_1. We construct a subgraph G'_1 of graph G_1 as follows:

- In the graph G_1, for each set I_1^i, keep only the vertex v_i^J where the vertex corresponding to v_i in G gets color $J \in [k]$ in the solution Π and delete other vertices of I_1^i
- Delete the edges of v_i^J to those vertices of B_1 that are adjacent to some vertex v_p^Q , where $p \leq i - 1$ and $Q \in [k]$. Now no two vertices v_i^J, v_p^Q have any common neighbor.
- Let n_i be the number of neighbors of v_i^J in B_1, for $i \in [t], J \in [k]$.
- If $n_i = 0$, then we also delete v_i^J

- After this step let $\{v_1^{J_1}, v_2^{J_2}, \dots v_h^{J_h}\}$ be the vertices remaining from vertices of the form v_i^J
- Now in A_1, B_1 keep only the vertices w_i, u_i such that u_i has neighbor of the form v_p^J.
- In the set I_1 keep only vertices v_p such that it has a neighbor of the form v_p^J. Let the remaining vertices be $\{v_1, \dots v_h\}$.
- Delete all the isolated K_{50}.

We will show that this constructed subgraph G_1' is isomorphic to a pattern graph $G_2 \in \mathcal{H}$ corresponding to the values $n_1 + n_2 + \dots + n_h = b$. We show the isomorphism $f : V(G_1') \to V(G_2)$ as follows.

- For every $v_i^{J_i} \in G_1'$, we map $f(v_i^{J_i}) = v_i'$.
- Now both $v_i^{J_i}, v_i'$ have n_i neighbors in B_1, B_2 respectively. Suppose $v_i^{J_i}$ is adjacent to $\{u_{i_1}, u_{i_2}, \dots u_{i_{n_i}}\} \subset B_1$. Then we have $f(u_{i_j}) = u'_{(1+\dots n_{i-1})+j}$
- Correspondingly $f(w_{i_j}) = w'_{(1+\dots n_{i-1})+j}$
- For $v_i \in I_1$ we have $f(v_i) = v_i''$.
- The K_{50} which is neighbor of $v_i \in I_1$ gets mapped to the K_{50} which is neighbor of v_i''
- K_{100} in G_1' gets mapped to the K_{100} in G_2

Conversely, suppose that there is a subgraph of G_1 that is isomorphic to some pattern graph $G_2 \in \mathcal{H}$. Since K_{100} is the largest clique in G_1, it is isomorphic to the K_{100} in G_2. The set I_2' in G_2 is isomorphic to a subset of the set I_1 in G_1, as the vertices of both sets are neighbors of the cliques K_{100} and copies of K_{50}. Every vertex $v_i'' \in I_2'$ is adjacent one vertex in I_2. Thus at most one vertex from each I_1^i is mapped to a vertex in I_2. Now for each set I_1^i consider the vertex that was mapped to a vertex in I_2. It has neighbors in B_1 and hence this subset of B_1 is mapped to B_2 in G_2. Similarly, the corresponding subset of A_1 is mapped to A_2 in G_2. Since, the pattern graph G_2 is isomorphic to a subgraph of G_1 and there exists at most one vertex v_i^J in each I_1^i in G_1 that has n_i private neighbors in B_1. Since $n_1 + \dots n_{|M|} = b$, the selected vertices in I_1^i in total have b neighbors in B_1, which implies that each vertex in B_1 has a unique neighbor in I_1^i. Thus, the neighborhood of the mapped vertices in I_1^i in B_1 is exactly B_1. An edge (u_r, v_i^J) in G_1 implies that the vertex v_i when colored with color J satisfies the bad pair corresponding to r in B_1. Thus, all the bad pairs in B_1 are being satisfied. Therefore, we can conclude that (G, S, k, ℓ, π) is a `YES` instance. ◀

▶ **Fact 2.** [1, Theorem 6.3] *Let H be a directed or an undirected graph on k vertices with treewidth tw. Let $G = (V, E)$ be a (directed or undirected) graph. A subgraph of G isomorphic to H, if one exists, can be found in the expected time $2^{\mathcal{O}(k)} \cdot |V|^{tw+1}$ and in the worst-case time $2^{\mathcal{O}(k)} \cdot |V|^{tw+1} \cdot \log |V|$.*

We describe our algorithm in Figure 4. Note that by the construction of our pattern graph and the proof of Lemma 24 our algorithm will also be able to return a satisfying coloring in the case of a `YES` instance.

Running time. First, we analyze the running time of the algorithm for Disjoint Achromatic Number/vc. The number of guesses in step 2 is at most $b2^b \le k^2 2^{k^2}$. The running time of the algorithm at step 4 is $2^{\mathcal{O}(k^2)} n^{\mathcal{O}(1)}$ from Fact 2. Thus, the overall running time of the algorithm for Disjoint Achromatic Number/vc is $k^2 2^{k^2} 2^{\mathcal{O}(k^2)} n^{\mathcal{O}(1)} = 2^{\mathcal{O}(k^2)} n^{\mathcal{O}(1)} = 2^{\mathcal{O}(\ell^2)} n^{\mathcal{O}(1)}$ since $\ell \le k+1$ from Observation 18. Note that the proof works even if the cliques in the host and pattern graph are K_3 and K_5 in place of K_{50} and K_{100}, respectively. We assumed a large clique for clarity of understanding. Now, in the algorithm for Achromatic Number/VC, we make ℓ^ℓ many guesses of the partitions of S and run the algorithm for

Disjoint Achromatic Number/vc ($I = (G, S, k, \pi)$)
(It checks if (G, S, k, π) is a `YES` instance)

1. Construct the host graph G_1.
2. Guess a value of $|M|$, ($|M| \leq b$), and a partition of b into $|M|$ positive integers $\{n_1, \dots, n_{|M|}\}$.
3. For each such guess construct a pattern graph G_2.
4. Use the algorithm from Fact 2 to check if a subgraph of G_1 is isomorphic to G_2 .
5. If a subgraph of G_1 is isomorphic to G_2 output `YES` instance .
6. Return `NO` if none of the pattern graphs G_2 is isomorphic to a subgraph of G_1

Figure 4 Algorithm for Disjoint Achromatic Number/vc.

Disjoint Achromatic Number/vc for each guess. Thus, the overall running time of the algorithm for Disjoint Achromatic Number/vc is $\ell^\ell \cdot 2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)} = 2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)}$. Thus, we get the following theorem.

▶ **Theorem 4.** Achromatic Number/VC *can be solved in* $2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)}$ *time.*

5 Kernel for d-degenerate graphs

In this section, we obtain a polynomial kernel for Achromatic Number when the graph is d-degenerate. First, we state a crucial theorem that we use to design the polynomial kernel for graphs that are d-degenerate. Then, we prove a few lemmatas to show the correctness of our kernelization algorithm. We start with the following lemma, which asserts that if a graph contains a *large* induced matching, then the achromatic number of the graph is also large.

▶ **Lemma 25.** *Let G be an undirected graph. For any positive integer k, if G has an induced matching of size at least k^2, then $\psi(G) \geq k$.*

Proof. Let G be an undirected graph and k be a positive integer. Let us assume that the graph G contains an induced matching M of size $\binom{k}{2}$. The set $V(M)$ denotes the set of vertices in G that are saturated by the matching M. First, we give a complete coloring using k colors on the matched vertices. Note that there are exactly $\binom{k}{2} \leq k^2$ distinct pairs of colors. We color the end points of the edges of M in the following greedy way. We say that a pair (i, j) is unassigned if there is no edge $e = (u, v)$ in M such that color $(u) = i$ and color $(v) = j$. Furthermore, an edge $e = (u, v)$ is uncolored if we have not assigned any color to u and v. Initially, all edges in M are uncolored, and all pairs (i, j), where $1 \leq i < j \leq [k]$, are unassigned. In each step, we take an uncolored edge $e = (u, v) \in M$, an unassigned pair (i, j), chosen arbitrarily, and add the color i and j to u and v, respectively. We stop when there is no such pair and an uncolored edge exists. As both the number of pairs and edges in M is exactly $\binom{k}{2}$, for any $i, j \in [k]$ with $i < j$, we have exactly one edge whose end points are colored with colors i and j.

In the above process, we color exactly $2|M|$, that is, $2\binom{k}{2}$ vertices. For the remaining $n - 2\binom{k}{2}$ vertices in the graph, we do the following. Let t denote the number of colors used so far in the process. Obviously, $t \geq k$. We take an uncolored vertex $v \in V(G) \setminus V(M)$. If there is an $i \in [t]$ such that v does not have neighbors in the vertices that have already been colored i, we assign the color i to the vertex v. If no such i exists, then we give a new color to the vertex v. It is easy to see that this procedure does not violate the complete coloring property. As we use at least k colors, we get a complete coloring of G with at least k colors. ◀

Kernelization Algorithm($G = (V, E), k$)
Given a d-degenerate graph G, it returns a kernel of size $k^{\mathcal{O}(d)}$.

1. Apply Reduction Rule 2 exhaustively to get G' with n' vertices.
2. If $|n'| \geq \frac{4k(k+1)}{3}(k^3 + 8d - 1)^{8d}$, return a trivial YES instance
3. Else, return the reduced graph G'.

Figure 5 Kernelization Algorithm for Achromatic Number on d-degenerate graphs.

▶ **Definition 26** (Strong system of distinct representatives). A *system of distinct representatives* for the sets $S_1, S_2, \ldots, S_k$ is a k-tuple $(x_1, x_2, \ldots, x_k)$ where the elements x_i are distinct and $x_i \in S_i$, for all $i \in [k]$. In addition to that, if we have $x_i \notin S_j$ for all $i \neq j$, then such a system is called *strong*.

▶ **Theorem 27** ([16, Theorem 8.12]). *For a pair of integers r and t, in any family of more than $\binom{r+t}{t}$ sets of cardinality at most r, at least $t+2$ of its members have a strong system of distinct representatives.*

The following property of a d-degenerate graph follows directly from the definition.

▶ **Proposition 28.** *The number of edges in a d-degenerate graph is bounded by dn.*

Next, we give a lower bound on the number of low-degree vertices in a d-degenerate graph.

▶ **Lemma 29.** *Let G be a d-degenerate graph and c be a positive integer with $c > 2$. Then, G contains strictly more than $(\frac{c-2}{c})n$ vertices with degree at most cd.*

Proof. Let G be a d-degenerate graph on n vertices. By Proposition 28, the number of edges is at most dn. Therefore, the sum of the degrees of the vertices in G is bounded by $2dn$. Assume that there are at most $(\frac{c-2}{c})n$ vertices of degree at most cd in G. Then we have a set $U \subseteq V(G)$ of size at least $n - (\frac{c-2}{c})n = \frac{2n}{c}$ vertices of degree strictly more than cd. Now, the sum of the degrees of the vertices in U is strictly more than $\frac{2n}{c} \cdot cd = 2dn$, a contradiction. Therefore, there are strictly more than $(\frac{c-2}{c})n$ vertices of degree at most cd in G. ◀

Let us define a *greedy independent partition* of G as follows. Construct a partition of $V(G)$ into independent sets by iteratively finding maximal independent sets. Then, in such a partition, among all sets, choose a maximal independent, say I_1, which has the highest cardinality (if there is more than one such set, then arbitrarily choose one) and arbitrarily order the remaining sets in the partition keeping I_1 as the first element. We denote this ordered family of sets as a *greedy independent partition*. Note that a *greedy independent partition* of G can be constructed in $\mathcal{O}(m)$ time.

Now, we describe our kernelization algorithm. A brief description of our algorithm is given in Figure 5.

▶ **Theorem 3.** Achromatic Number *admits a kernel of size $\mathcal{O}(k^{24d+2})$ on d-degenerate graphs.*

Proof. First, we construct a *greedy independent partition* of G, say $I_1, I_2, \ldots, I_p$. If $p \geq k$, then from Lemma 17, we get a complete coloring of G of size at least k and return a trivial YES-instance. Else, we do as follows.

We apply Reduction Rule 2 exhaustively. Let G' be the reduced graph with n' vertices. Note that after this step, at most $k+1$ vertices can have the same open neighborhood. Since our graph G is d-degenerate, the reduced graph G' is also d-degenerate. From Lemma 29, we know that the number of vertices with degree at most $8d$ is at least $\frac{3n'}{4}$. Let us call vertices with degree at most $8d$ as low-degree vertices. We denote these vertices by V_{low}. Let $\mathcal{A} = \{A_1, A_2, \ldots A_l\}$ be a *greedy independent partition* of $G'[V_{\text{low}}]$. If $|A_1| < \frac{3n'}{4k}$, then we have $l \geq k$ (as $|V_{\text{low}}| \geq \frac{3n'}{4}$). So, applying Lemma 17, we have $\psi(G'[V_{\text{low}}]) \geq k$ and that immediately implies $\psi(G') \geq k$, and we return a trivial YES instance. Otherwise, we have $|A_1| \geq \frac{3n'}{4k}$.

Let $\{v_1, v_2, \ldots, v_{|A_1|}\}$ be the vertices in A_1 and $\{S_1, S_2, \ldots, S_{|A_1|}\}$ be their corresponding open neighborhoods. Due to the exhaustive application of Reduction Rule 2 at least $\frac{|A_1|}{k+1}$ of these open neighborhoods are distinct. Let $\frac{|A_1|}{k+1} = q$ and $\{S_1, S_2, \ldots, S_q\}$ be the pairwise distinct sets. Let $S = \bigcup_{i=1}^{q} \{S_i\}$. Thus, S is a family of more than $\frac{3n'}{4k(k+1)}$ sets, each of cardinality at most $8d$. Let k' be the integer such that

$$\binom{8d+k'-1}{k'-1} \leq \frac{3n}{4k(k+1)} \leq \binom{8d+k'}{k'} \tag{2}$$

By applying Theorem 27 on the family of sets S, we know at least $k'-1+2 = k'+1$ of its members have a strong system of distinct representatives. We match each representative vertex with its corresponding vertex in A_1. Let B be the set of those representative vertices. Basically B represents those set of vertices in $G - A_1$ such that we have a set of vertices $A' \subseteq A_1$ satisfying (i) $|B| = |A'|$, (ii) for each vertex $b \in B$ there exists a unique vertex $v_a \in A'$ such that $b \in N(v_a)$ and $b \notin N(v)$ for all $v \in A' \setminus \{v_a\}$. We compute a *greedy independent partition*, say $\mathcal{B} = \{B_1, B_2, \ldots, B_j\}$ in $G'[B]$. Now we have the following cases.

- If $|B_1| < \frac{|B|}{k}$, then we have $j \geq k$. From Lemma 17, we get $\psi(G'[B]) \geq k$ and that immediately implies $\psi(G') \geq k$, and we return a trivial YES instance.
- Otherwise, we have $|B_1| \geq \frac{|B|}{k}$. So, we have an induced matching of size at least $\frac{|B|}{k}$ in G'.
  - If $\frac{|B|}{k} \geq k^2$, then we obtain an induced matching of size at least k^2. Then applying Lemma 25, we have $\psi(G') \geq k$. Hence, we return a trivial YES instance.
  - Otherwise, we have $\frac{|B|}{k} < k^2$.

Now $|B| \geq k'+1$ and $\frac{|B|}{k} < k^2$ together imply

$$k' \leq k^3 - 1 \tag{3}$$

Using a binomial formula, we have the following.

$$\binom{8d+k'}{k'} = \binom{8d+k'}{8d} \leq (8d+k')^{8d} \tag{4}$$

From Equation (2) and Equation (4) we have

$$\frac{3n'}{4k(k+1)} \leq \binom{8d+k'}{k'} \leq (8d+k')^{8d}$$

$$\Rightarrow (8d+k') \geq (\frac{3n'}{4k(k+1)})^{\frac{1}{8d}}$$

$$\Rightarrow k' \geq (\frac{3n'}{4k(k+1)})^{\frac{1}{8d}} - 8d \tag{5}$$

From Equation (3) and Equation (5), we get

$$
\begin{aligned}
(\frac{3n'}{4k(k+1)})^{\frac{1}{8d}} - 8d \leq k' < k^3 - 1 & \\
n' < \frac{4k(k+1)}{3}(k^3 + 8d - 1)^{8d} & \\
n' = \mathcal{O}(k^{24d+2}) & \qquad (6)
\end{aligned}
$$

Hence Achromatic Number admits a kernel of size $\mathcal{O}(k^{24d+2})$ on d-degenerate graph. ◀

6 Conclusion

In this paper, we do a parameterized reunion with Achromatic Number, and design an FPT algorithm with explicit running time on general graphs. We also study the problem with respect to structural parameterizations. Our work leaves several intriguing open questions.

1. Does Achromatic Number admit a polynomial kernel on general graphs?
2. We showed that Achromatic Number/VC can be solved in $2^{\mathcal{O}(\ell^2)} \cdot n^{\mathcal{O}(1)}$. We can show the following lower bound on the running time of an algorithm for Achromatic Number/VC.

   ▶ **Theorem 30** (♠). *Unless* ETH *fails,* Achromatic Number/VC *cannot be solved in time* $2^{o(\ell)} \cdot n^{\mathcal{O}(1)}$*, where* ℓ *is the size of the vertex cover.*

   A natural question is whether we could close the gap between the upper and the lower bounds on the running time of Achromatic Number/VC.

References

1 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *Journal of the ACM (JACM)*, 42(4):844–856, 1995. doi:10.1145/210332.210337.

2 Katerina Asdre, Kyriaki Ioannidou, and Stavros D. Nikolopoulos. The harmonious coloring problem is np-complete for interval and permutation graphs. *Discret. Appl. Math.*, 155(17):2377–2382, 2007. doi:10.1016/J.DAM.2007.07.005.

3 V. Bhave. On the pseudoachromatic number of a graph. *Fundamenta Mathematicae*, 102(3):159–164, 1979.

4 Hans L. Bodlaender. Achromatic number is np-complete for cographs and interval graphs. *Inf. Process. Lett.*, 31(3):135–138, 1989. doi:10.1016/0020-0190(89)90221-4.

5 Béla Bollobás, Bruce A. Reed, and Andrew Thomason. An extremal function for the achromatic number. In Neil Robertson and Paul D. Seymour, editors, *Graph Structure Theory, Proceedings of a AMS-IMS-SIAM Joint Summer Research Conference on Graph Minors held June 22 to July 5, 1991, at the University of Washington, Seattle, USA*, volume 147 of *Contemporary Mathematics*, pages 161–165. American Mathematical Society, 1991.

6 Niall Cairnie and Keith Edwards. Some results on the achromatic number. *J. Graph Theory*, 26(3):129–136, 1997. doi:10.1002/(SICI)1097-0118(199711)26:3\%3C129::AID-JGT3\%3E3.0.CO;2-T.

7 Jianer Chen, Iyad A. Kanj, Jie Meng, Ge Xia, and Fenghui Zhang. On the pseudo-achromatic number problem. *Theor. Comput. Sci.*, 410(8-10):818–829, 2009. doi:10.1016/J.TCS.2008.11.010.

8 K. Edwards and C. McDiarmid. The complexity of harmonious colouring for trees. *Discrete Applied Mathematics*, 57(2-3):133–144, 1995. Copyright 2009 Elsevier B.V., All rights reserved. doi:10.1016/0166-218X(94)00100-R.

9 Keith J. Edwards. Achromatic number versus pseudoachromatic number: a counterexample to a conjecture of hedetniemi. *Discrete Mathematics*, 219(1):271–274, 2000. doi:10.1016/S0012-365X(00)00025-X.

10 Martin Farber, Gena Hahn, Pavol Hell, and Donald J. Miller. Concerning the achromatic number of graphs. *J. Comb. Theory, Ser. B*, 40(1):21–39, 1986. doi:10.1016/0095-8956(86)90062-6.

11 John P. Georges. On the harmonious coloring of collections of graphs. *Journal of Graph Theory*, 20(2):241–254, 1995. doi:10.1002/jgt.3190200213.

12 R. P. Gupta. The achromatic number of a graph. *Journal of Combinatorial Theory*, 1969.

13 Frnak Harary and Stephen Hedetniemi. Bounds on the chromatic and achromatic numbers of complementary graphs. *Recnet Progress in Combinatorics*, 8(2):154–161, 1970. doi:10.1016/S0021-9800(70)80072-2.

14 Pavol Hell and Donald J. Miller. Graph with given achromatic number. *Discret. Math.*, 16(3):195–207, 1976. doi:10.1016/0012-365X(76)90099-6.

15 J. Hopcroft and Mukkai Krishnamoorthy. On the harmonious coloring of graphs. *Siam Journal on Algebraic and Discrete Methods*, 4, September 1983. doi:10.1137/0604032.

16 Stasys Jukna. *Extremal Combinatorics – With Applications in Computer Science*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2011. doi:10.1007/978-3-642-17364-6.

17 Sudeshna Kolay, Ragukumar Pandurangan, Fahad Panolan, Venkatesh Raman, and Prafullkumar Tale. Harmonious coloring: Parameterized algorithms and upper bounds. *Theor. Comput. Sci.*, 772:132–142, 2019. doi:10.1016/J.TCS.2018.12.011.

18 Guy Kortsarz and Robert Krauthgamer. On Approximating the Achromatic Number. *SIAM J. Discret. Math.*, 14(3):408–422, 2001. doi:10.1137/S0895480100379579.

19 Attila Máté. A lower estimate for the achromatic number of irreducible graphs. *Discret. Math.*, 33(2):171–183, 1981. doi:10.1016/0012-365X(81)90164-3.

20 Z. Miller and D. Pritikin. The harmonious coloring number of a graph. *Discrete Mathematics*, 93(2):211–228, 1991. doi:10.1016/0012-365X(91)90257-3.

21 E. Sampathkumar. Partition graphs and coloring numbers of a graph. *Discrete Mathematics – DM*, 16:57–60, September 1976. doi:10.1016/0012-365X(76)90093-5.

22 Mihalis Yannakakis and Fanica Gavril. Edge Dominating Sets in Graphs. *SIAM J. Appl. Math.*, 38:364–372, June 1980. doi:10.1137/0138030.