Parameterized Saga of First-Fit and Last-Fit Coloring

Akanksha Agrawal
Indian Institute of Technology Madras, India

Daniel Lokshtanov
University of California, Santa Barbara,CA, USA

Fahad Panolan
School of Computer Science, University of Leeds, UK

Saket Saurabh
The Institute of Mathematical Sciences, HBNI, Chennai, India
Department of Informatics, University of Bergen, Norway

Shaily Verma
Algorithm Engineering Group, Hasso Plattner Institute, Potsdam, Germany

Abstract

The classic greedy coloring algorithm considers the vertices of an input graph G in a given order and assigns the first available color to each vertex v in G. In the Grundy Coloring problem, the task is to find an ordering of the vertices that will force the greedy algorithm to use as many colors as possible. In the Partial Grundy Coloring, the task is also to color the graph using as many colors as possible. This time, however, we may select both the ordering in which the vertices are considered and which color to assign the vertex. The only constraint is that the color assigned to a vertex v is a color previously used for another vertex if such a color is available.

Whether Grundy Coloring and Partial Grundy Coloring admit fixed-parameter tractable (FPT) algorithms, algorithms with running time $f(k)n^{\mathcal{O}(1)}$, where k is the number of colors, was posed as an open problem by Zaker and by Effantin et al., respectively.

Recently, Aboulker et al. (STACS 2020 and Algorithmica 2022) resolved the question for Grundy Coloring in the negative by showing that the problem is W[1]-hard. For Partial Grundy Coloring, they obtain an FPT algorithm on graphs that do not contain $K_{i,j}$ as a subgraph (a.k.a. $K_{i,j}$-free graphs). Aboulker et al. re-iterate the question of whether there exists an FPT algorithm for Partial Grundy Coloring on general graphs and also asks whether Grundy Coloring admits an FPT algorithm on $K_{i,j}$-free graphs. We give FPT algorithms for Partial Grundy Coloring on general graphs and for Grundy Coloring on $K_{i,j}$-free graphs, resolving both the questions in the affirmative. We believe that our new structural theorems for partial Grundy coloring and "representative-family" like sets for $K_{i,j}$-free graphs that we use in obtaining our results may have wider algorithmic applications.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Grundy Coloring, Partial Grundy Coloring, FPT Algorithm, $K_{i,j}$-free graphs

Digital Object Identifier 10.4230/LIPIcs.STACS.2025.5

Related Version *Full Version*: https://arxiv.org/pdf/2410.20629

Funding *Akanksha Agrawal*: Supported by the New Faculty Seed Grant, IIT Madras (No. NFSC008972).
Saket Saurabh: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416).

42nd International Symposium on Theoretical Aspects of Computer Science (STACS 2025).
Editors: Olaf Beyersdorff, Michał Pilipczuk, Elaine Pimentel, and Nguyễn Kim Thắng;
Article No. 5; pp. 5:1–5:21
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

A *proper coloring* of a graph G is an assignment of colors to its vertices such that none of the edges have endpoints of the same color. In k-Coloring, we are given a graph G, and the objective is to test whether G admits a proper coloring using at most k colors. The k-Coloring problem is one of the classical NP-hard problems, and it is NP-complete for every fixed $k \geq 3$. The problem is notoriously hard even to approximate. Indeed, approximating k-Coloring within $\mathcal{O}(n^{1-\epsilon})$, for any $\epsilon > 0$, is hard [18]. Also, under a variant of Unique Games Conjecture, there is no constant factor approximation for 3-Coloring [14].

The k-Coloring problem has varied applications ranging from scheduling, register allocations, pattern matching, and beyond [8, 9, 29]. Owing to this, several heuristics-based algorithms have been developed for the problem. One of the most natural greedy strategies considers the vertices of an input graph G in an arbitrary order and assigns to each vertex the first available color in the palette (the color palette for us is $\mathbb{N}$). In literature, this is called the *first-fit* rule. Notice that there is nothing special about using the "first" available color; one may instead opt for any of the previously used colors, if available, before using a new color; let us call this greedy rule the *any-available* rule. It leads to yet another greedy strategy to properly color a graph, and one can easily prove that this greedy strategy is equivalent to the "last-(available) fit" rule.

For any greedy strategy, one may wonder: *How bad can the strategy perform for the given instance?* The above leads us to the well-studied fundamental combinatorial problems, Grundy Coloring and Partial Grundy Coloring, that arise from the aforementioned greedy strategies for proper coloring. In the Grundy Coloring problem, we are given a graph G on n vertices and an integer k, and the goal is to check if there is an ordering of the vertices on which the first-fit greedy algorithm for proper coloring uses at least k colors. Similarly, we can define the Partial Grundy Coloring problem, where the objective is to check if, for the given graph G on n vertices and integer k, there is an ordering of the vertices on which the any-available greedy algorithm uses at least k colors. In this paper, we consider these two problems in the realm of parameterized complexity.

The Grundy Coloring problem has a rich history dating back to 1939 [21]. Goyal and Vishwanathan [20] proved that Grundy Coloring is NP-hard. Since then, there has been a flurry of results on the computational and combinatorial aspects of the problem both on general graphs and on restricted graph classes, see, for instance [3, 6, 7, 10, 16, 23, 24, 26, 27, 33, 35–39] (this list is only illustrative, not comprehensive). The problem Partial Grundy Coloring was introduced by Erdös et al. [16] and was first shown to be NP-hard by Shi et al. [34]. The problem has gained quite some attention thereafter; see, for instance [1, 4, 5, 12, 15, 25, 32, 36].

These problems have also been extensively studied from the parameterized complexity perspective. Unlike k-Coloring, both these problems admit XP algorithms [15, 38], i.e., an algorithm running in time bounded by $|V(G)|^{f(k)}$. The above naturally raises the question of whether they are fixed-parameter tractable (FPT), i.e., admit an algorithm running in time $f(k) \cdot |V(G)|^{\mathcal{O}(1)}$. In fact, these problems have also been explicitly stated as open problems [1, 7, 23, 33].

Havet and Sampaio [23] studied Grundy Coloring with an alternate parameter and showed that the problem of testing whether there is a Grundy coloring with at least $|V(G)| - q$ colors is FPT parameterized by q. Bonnet et al. [7] initiated a systematic study of designing parameterized and exact exponential time algorithms for Grundy Coloring and obtained FPT algorithms for the problem for several structured graph classes. They gave an exact algorithm for Grundy Coloring running in time $2.443^n \cdot n^{\mathcal{O}(1)}$ and also showed that the problem is FPT on chordal graphs, claw-free graphs and graphs excluding a fixed minor. In the same paper, they stated the tractability status of Grundy Coloring on general graphs parameterized by the treewidth or the number of colors as central open questions. Belmonte et al. [6] resolved the first question by proving that Grundy Coloring is W[1]-hard parameterized by treewidth, but surprisingly, it becomes FPT parameterized by pathwidth. Later, Aboulker et al. [1] proved that Grundy Coloring does not admit an FPT algorithm (parameterized by the number of colors) and obtained an FPT algorithm for Partial Grundy Coloring on $K_{t,t}$-free graphs (which includes graphs of bounded degeneracy, graphs excluding some fixed graph as minor/topological minors, graphs of bounded expansion and nowhere dense graphs). A graph is $K_{i,j}$-free if it does not have a *subgraph* isomorphic to the complete bipartite graph with i and j vertices, respectively, on the two sides. They concluded their work with the following natural open questions:

Question 1: Does Partial Grundy Coloring admit an FPT algorithm?
Question 2: Does Grundy Coloring admit an FPT algorithm on $K_{i,j}$-free graphs?

In this paper, we resolve the questions 1 and 2 in the affirmative by a new structural result and a new notion of representative families for $K_{i,j}$-free graphs, respectively. In the next section, we give an intuitive overview of both results, highlighting our difficulties and our approaches to overcome them.

1.1 Our Results, Methods and Overview

Our first result is the following.

▶ **Theorem 1.** *Partial Grundy Coloring is solvable in time* $2^{\mathcal{O}(k^5)} \cdot n^{\mathcal{O}(1)}$.

Our algorithm starts with the known "witness reformulation" of Partial Grundy Coloring. It is known that (G, k) is a yes-instance of Partial Grundy Coloring if and only if there is a vertex subset W of size at most k^2 such that, $(G[W], k)$ is a yes-instance of the problem. In the above, the set W is known as a *small witness set*. Our algorithm is about finding such a set W of size at most k^2. Observe that this witness reformulation immediately implies that Partial Grundy Coloring admits an algorithm with running time $n^{\mathcal{O}(k^2)}$ time. To build our intuition, we first give a simple algorithm for the problem on graphs of bounded degeneracy (or even, nowhere dense graphs). This algorithm has two main steps: (a) classical color-coding of Alon-Yuster-Zwick [2], and (b) independence covering lemma of Lokshtanov et al. [28].

Let (G, k) be a yes instance of Partial Grundy Coloring, where G is a d-degenerate graph on n vertices, and W be a small witness set of size at most k^2. As $(G[W], k)$ must be a yes-instance of the problem, there exists an ordering of the vertices such that when we apply *any-available* greedy rule, it uses at least k colors. Let $\widehat{c}$ be this proper coloring of $G[W]$. The tuple $(W_i := \{v \in W \mid \widehat{c}(v) = i\})_{i \in [k]}$ is called a *k-partial Grundy witness* for G. Now we apply the color-coding step of the algorithm. That is, we color the vertices of G using k colors independently and uniformly at random, and let $Z_1, \cdots, Z_k$ be the color classes of this coloring. The probability that for each $i \in [k]$, $W_i \subseteq Z_i$, is k^{-k^2}. Notice that since G is a d-degenerate graph, we have that $G_i = G[Z_i]$, for each $i \in [k]$, is d-degenerate. Now we exploit this fact and apply the independence covering lemma of Lokshtanov et al. [28]. That is given as input (G_i, k^2), in time $2^{\mathcal{O}(dk^2)} n^{\mathcal{O}(1)}$ it produces a family $\mathcal{F}_i$ of independent sets of G_i, of size $2^{\mathcal{O}(dk^2)} \cdot \log n$. Furthermore, given any independent set I of G_i of size at most k^2, there exists an independent set $F \in \mathcal{F}_i$, such that $I \subseteq F$ (F covers I). In particular, we know that there is a set $F_i \in \mathcal{F}_i$ that covers W_i. So, the algorithm just enumerates each tuple $(F_1, \ldots, F_k)$ of $\mathcal{F}_1 \times \cdots \times \mathcal{F}_k$ and checks whether $(F_1, \cdots, F_k)$ is a k-partial Grundy coloring of $G[F_1 \cup \cdots \cup F_k]$. If (G, k) is a yes instance, our algorithm is successful with probability k^{-k^2}. Moreover, we can convert the described randomized algorithm to a deterministic one by using the standard derandomization technique of "universal sets" [17, 31]. Some remarks are in order; it can be shown that each of $|W_i| \leq k$, and hence we can call the independence covering lemma on (G_i, k), resulting in an improved running time of $2^{\mathcal{O}(dk^2 \log k)} \cdot n^{\mathcal{O}(1)}$. Aboulker et al. [1] proved that Partial Grundy Coloring on $K_{t,t}$-free graphs is FPT, which includes t-degenerate graphs. Aboulker et al. did not explicitly mention the running time, but their running time is at least $2^{k^t} n$. Our new algorithm improves over this. For general instances, we do not have a bound on the degeneracy of the input graph. However, we can achieve this by using our new structural result.

▶ **Theorem 2** (Structural result). *There is a polynomial-time algorithm that given a graph G and a positive integer k, does one of the following:*

(i) *Correctly concludes that (G, k) is a yes-instance of Partial Grundy Coloring, or*

(ii) *Outputs at most $2k^3$ induced bicliques $A_1, \cdots, A_\ell$ in G such that the following holds. For any $v \in V(G)$, the degree of v in $G - F$ is at most k^3, where F is the union of the edges in the above bicliques.*

The structural result (Theorem 2) is one of our main technical contributions. Next, we show how to design an algorithm for Partial Grundy Coloring using Theorem 2. We follow the same steps as for the one described for the degenerate case. That is, we have color classes Z_is and they contain the respective W_is (the part of the small witness set W). Now, to design a family of independent sets in $G_i = G[Z_i]$, we do as follows. Let (L_j, R_j) be a bipartition of A_j, for each $j \in [\ell]$. Observe that any independent set I (in particular of G_i) intersects L_j or R_j, but *not both*, for any $j \in [\ell]$. Thus, we first guess whether W_i intersects L_j, R_j or none. Let this be given by a function $f_i : [\ell] \to \{L, R, N\}$, that is, if $f_i(j) = L$, then $W_i \cap L_j = \emptyset$, if $f_i(j) = R$, then $W_i \cap R_j = \emptyset$, else $W_i \cap (L_j \cup R_j) = \emptyset$. Taking advantage of this property, for each guess of which of L_j or R_j is not contained in W_i, we delete the corresponding set (which is one of L_j or R_j, for each $j \in [\ell]$) from G_i. We call the resulting graph $G_i^{f_i}$. This implies that for any f_i, in $G_i^{f_i}$ we delete all edges of F (where F is the union of edges in the bicliques). Hence, the maximum degree of $G_i^{f_i}$ is at most k^3, and therefore it has degeneracy at most k^3. Now using the independence covering step of the algorithm for degenerate graphs, we can finish the algorithm. The proof of Theorem 2 is obtained by carefully analyzing the reason for the failure of a greedy algorithm.

Our next result is an affirmative answer to Question 2.

▶ **Theorem 3.** *For any fixed $i, j \in \mathbb{N}$, there is an FPT algorithm that given a graph G and a positive integer k, decides if there is Grundy coloring of G using at least k colors.*

For our algorithm, we use a reinterpretation of the problem which is based on the existence of a *small witness*. Gyárfás et al. [22], and Zaker [38] independently showed that a given instance (G, k) of Grundy Coloring is a yes-instance if and only if there is a vertex subset W of size at most 2^{k-1}, such that $(G[W], k)$ is also a yes-instance of the problem. The existence of this small induced subgraph directly yield an XP algorithm for the problem [38]. Using characterizations of [22,38] and basic Grundy coloring properties, we can reduce Grundy Coloring to finding a homomorphic image, satisfying some independence constraints, of some *specific labeled trees* (see Fig. 1, where different parts of it will be

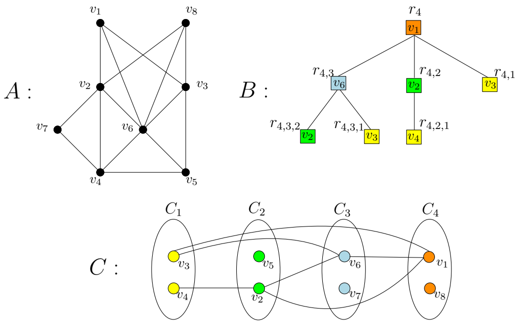

Figure 1 Illustration of a labelled homomorphism $\omega : V(T_4) \to V(G)$, where the graph is shown in part (i), T_4 with $V(T_4) = \{r_4, r_{4,3}, r_{4,2}, r_{4,1}, r_{4,3,2}, r_{4,3,1}, r_{4,2,1}\}$ is shown in part (ii), and a relation to Grundy coloring is illustrated in part (iii). Here, $\omega(r_4) = v_1$, $\omega(r_{4,3}) = v_6$, $\omega(r_{4,2}) = v_2, \omega(r_{4,1}) = v_3, \omega(r_{4,3,2}) = v_2, \omega(r_{4,3,1}) = v_3$, and $\omega(r_{4,2,1}) = v_4$.

discussed, shortly). Let the pair (T, ℓ) denote a rooted tree T together with a labeling function $\ell : V(T) \to [k]$. Given (T, ℓ) and a graph G, a function $\omega : V(T) \to V(G)$ is a *labeled homomorphism* if: i) for each $\{u, v\} \in E(T)$, we have $\{\omega(u), \omega(v)\} \in E(G)$, and ii) for $u, v \in V(T)$, if $\ell(u) \neq \ell(v)$, then $\omega(u) \neq \omega(v)$. In particular, we will reduce the problem to the following.

> Constrained Label Tree Homomorhism (CLTH) **Parameter:** $|V(T)|$
> **Input:** A host graph G and $(T, \ell : V(T) \to [k])$, where T is a tree.
> **Question:** Does there exists a labeled homomorphism $\omega : V(T) \to V(G)$ such that for any $z \in [k]$, $W_i = \{\omega(t) \mid t \in V(T) \text{ and } \ell(t) = i\}$ is an independent set in G?

Now our goal is to identify ω (and thus the witness set W). The first step of our algorithm will be to use the color-coding technique of Alon-Yuster-Zwick [2] to ensure that the labeling requirement of the graph homomorphism ω is satisfied. To this end, we randomly color the vertices of G using k colors, where we would like the random coloring to ensure that for each $z \in [k]$, all the vertices in W_z are assigned the color z. Let $X_1, X_2, \cdots, X_k$ be the color classes in a coloring that achieves the above property. Our objective will be to find ω such that vertices of T that are labeled $z \in [k]$ are assigned to vertices in X_z.

Our next challenge is to find a homomorphism that additionally satisfies the independence condition. That is, vertices of the same label in T are assigned to an independent set in G. Note that the number of potential ωs that satisfy our requirements can be very huge; however, we will be able to carefully exploit $K_{i,j}$-freeness to design a dynamic programming-based algorithm to identify one such ω (and thus the set W). Our approach here is inspired by dynamic programming in the design of FPT algorithms based on computations of "representative sets" [19,30]. However, this inspiration ends here, as to apply known methods we need to have an underlying family of sets that form a matroid. Unfortunately, we do not have any matroid structure to apply the known technique. Here, we exploit the fact that we have a specific labeled tree (T, ℓ) and a $K_{i,j}$-free graph. Next we define the specific trees that we will be interested in (see Fig. 1, (ii)).

▶ **Definition 4.** For each $k \in \mathbb{N} \setminus \{0\}$, we (recursively) define a pair $(T_k, \ell_k : V(T_k) \to [k])$, called a *k-Grundy tree*, where T_k is a tree and ℓ_k is a *labelling* of $V(T_k)$, as follows :

1. $T_1 = (\{r_1\}, \emptyset)$ is a tree with exactly one vertex r_1 (which is also its root), and $\ell_1(r_1) = 1$.
2. Consider any $k \geq 2$, we (recursively) obtain T_k as follows. For each $z \in [k-1]$, let (T_z, ℓ_z) be the z-Grundy tree with root r_z. We assume that for distinct $z, z' \in [k-1]$, T_z and

$T_{z'}$ have no vertex in common, which we can ensure by renaming the vertices.[1] We set $V(T_k) = \big(\cup_{z\in[k-1]} V(T_z)\big) \cup \{r_k\}$ and $A(T_k) = \big(\cup_{z\in[k-1]} A(T_z)\big) \cup \{(r_k, r_z) \mid z \in [k-1]\}$.
We set $\ell_k(r_k) = k$, and for each $z \in [k-1]$ and $t \in V(T_z)$, we set $\ell_k(t) = \ell_z(t)$.
For $v \in V(T_k)$, $\ell_k(v)$ is the *label* of v in (T_k, ℓ_k), and the elements in $[k]$ are *labels* of T_k.

Observe that the label of a vertex $t \in V(T)$ is the depth of the subtree rooted at t (the depth of a leaf is 1). In particular, the leaves are assigned the label 1, and when they are deleted, we get vertices with the label 2 as leaves, and so on. This allows us to do a bottom-up dynamic programming over T_k. Roughly speaking, for each $z \in [k]$, we solve the special labelled tree homomorphism from $\omega_z : V(T_z) \to X_1 \cup X_2 \ldots \cup X_z$, where the root of T_z is mapped to a fixed vertex $v \in X_z$ as follows: instead of having all potential choices for ω_z (or $W_z = \{\omega_z(t) \mid t \in V(T_z)\}$), we find enough representatives, that will allow us to replace W_z by something that we have stored. It is a priori not clear that such representative sets of small size exist and furthermore, even if they exist, how to find them. The existence and computation of small representative sets in this setting is our main technical contribution for Grundy Coloring.

We heavily exploit the $K_{i,j}$-freeness in our "representative set" computation. Very roughly stating, while we have computed required representatives for W_z, and wish to build such a representatives for W_{z+1}, by exploiting $K_{i,j}$-freeness, we either find a small hitting set or a large sub-family of pair-wise disjoint sets. In the former case, we can split the family and focus on the subfamily containing a particular vertex from the hitting set and obtain a "representative" for it (and then take the union over such families). In the latter case, we show that we are very close to satisfying the required property, except for the sets containing vertices from an appropriately constructed small set S of vertices. The construction of this *small* set S is crucially based on the $K_{i,j}$-freeness of the input graph. Once we have the set S, we can focus on sets containing a vertex from it and compute "representatives" for them.

Again, using standard hash functions, we can obtain a deterministic FPT algorithm for the problem by derandomizing the color coding based step [2, 31].

2 Preliminaries

Generic Notations. We denote the set of natural numbers by $\mathbb{N}$. For $n \in \mathbb{N}$, $[n]$ denotes the set $\{1, 2, \cdots, n\}$. For a function $f : X \to Y$ and $y \in Y$, $f^{-1}(y) := \{x \in X \mid f(x) = y\}$.

For standard graph notations not explicitly stated here, we refer to the textbook of Diestel [13]. For a graph G, we denote its vertex and edge set by $V(G)$ and $E(G)$, respectively. Also, if the context is clear, we will use n and m to denote the numbers $|V(G)|$ and $|E(G)|$, respectively. The neighborhood of a vertex v in a graph G is the set of vertices that are adjacent to v in G, and we denote it by $N_G(v)$. The degree of a vertex v is the size of its neighborhood in G, and we denote it by $d_G(v)$. For a set of vertices $S \subseteq V(G)$, we define $N_G(S) = (\cup_{v\in S} N(v)) \setminus S$. When the graph is clear from the context, we drop the subscript G from the above notations. For $X \subseteq V(G)$, the induced subgraph of G on X, denoted by $G[X]$, is the graph with vertex set X and edge set $\{\{u, v\} \mid u, v \in X \;\&\; \{u, v\} \in E(G)\}$. Also, $G[V(G) \setminus X]$ is denoted by $G - X$. For $v \in V(G)$, we use $G - v$ to denote $G - \{v\}$ for ease of notation. For an edge subset $F \subseteq E(G)$, $G - F$ is the graph with vertex set $V(G)$ and edge set $E(G) \setminus F$. A bipartite graph $G = (A \uplus B, E)$ is called a *biclique* if every vertex in A

[1] For the sake of notational simplicity we will not explicitly write the renaming of vertices used to ensure pairwise vertex disjointness of the trees. This convention will be followed in the relevant section.

is adjacent to every vertex in B. We assume that A and B are both non-empty sets. For $d \in \mathbb{N}$, a graph is *d-degenerate* if each of its subgraphs has a vertex of degree at most d. For terminologies related to parameterized complexity, see the textbook of Cygan et al. [11].

3 FPT Algorithm for Partial Grundy Coloring

Consider a graph G and an integer $k \in \mathbb{N} \setminus \{0\}$. For a (not necessarily proper) coloring $\chi : V(G) \to [k]$, for simplicity, we sometimes write χ as the ordered tuple $(\chi^{-1}(1), \chi^{-1}(2), \cdots, \chi^{-1}(k))$. Recall that a proper coloring of a graph is a coloring of its vertices so that for none of its edges, the two endpoints of it are of the same color. Also, a k-partial Grundy coloring of G is a proper coloring $c : V(G) \to [k]$, such that for each $i \in [k]$, there is a vertex $v \in V(G)$ with: (i) $c(v) = i$ and (ii) for every $j \in [i-1]$, there is $u \in N_G(v)$ with $c(u) = j$.

We will begin with some definitions and results that will be useful in obtaining our main structural result (Theorem 2) and our FPT-algorithm (Theorem 1).

Observe that given a k-partial Grundy coloring of an induced subgraph $\widehat{G}$ of a graph G, we can extend this coloring to a partial Grundy coloring of the whole graph G using at least k colors by greedily coloring the uncolored vertices of $G - V(\widehat{G})$. The following observation will be particularly useful when we work with a small "witness".

▶ **Observation 5.** *Given a graph G, an induced subgraph $\widehat{G}$ of G, and a partial Grundy coloring of $\widehat{G}$ using k colors, we can find a partial Grundy coloring of G using at least k colors in linear time.*

Proof. Let $\widehat{c} : V(\widehat{G}) \to [k]$ be a partial Grundy coloring of $\widehat{G}$ with exactly k colors. We construct a partial Grundy coloring $c : V(G) \to \mathbb{N}$ of G using at least k colors as follows. For each vertex $v \in V(\widehat{G})$, set $c(v) := \widehat{c}(v)$. Let $v_1, v_2, \cdots, v_{n'}$ be an arbitrarily fixed order of vertices in $V(G) \setminus V(\widehat{G})$. Let $G_0 = \widehat{G}$, and for each $p \in [n']$, $G_p = G[V(\widehat{G} \cup \{v_1, v_2, \cdots, v_p\})]$. We iteratively create a partial Grundy coloring c_p of G_p using at least k colors (in increasing values of p) as follows. Note that $c_0 = \widehat{c}$ is already a partial Grundy coloring of G_0 that uses at least k colors. Consider $p \in [n'] \setminus \{0\}$, and assume that we have already computed a partial Grundy coloring $c_{p-1} : V(G_{p-1}) \to \mathbb{N}$ of G_{p-1} that uses $k' \geq k$ colors. For each $z \in [k']$, let $V_z = c_{p-1}^{-1}(z)$. For each $v \in V(G_{p-1})$, we set $c_p(v) := c_{p-1}(v)$. If the vertex v_p has a neighbor in each of the sets $V_1, V_2, \cdots, V_{k'}$, i.e., if for each $z \in [k']$, $N_G(v_p) \cap V_z \neq \emptyset$, then set $c_p(v_p) := k' + 1$. Otherwise, let $z^* \in [k']$ be the smallest number such that $N_G(v_p) \cap V_{z^*} = \emptyset$, and set $c_p(v_p) := z^*$. Notice that by construction, c_p is a partial Grundy coloring of G_p using at least k colors. From the above discussions, $c_{n'}$ is a partial Grundy coloring of $G = G_{n'}$ using at least k colors. ◀

▶ **Definition 6.** Consider a graph G and an integer $k \in \mathbb{N} \setminus \{0\}$. A sequence of pairwise disjoint independent sets $(Q_1, Q_2, \ldots, Q_k)$ of G is a *k-partial Grundy witness* if the following holds. For any $i \in [k]$, there is $v \in Q_i$ such that for all $j \in [i-1]$, $Q_j \cap N_G(v) \neq \emptyset$. The vertex v is called a *dominator* in Q_i.

▶ **Observation 7.** *Given a graph G and an integer k, let $(Q_1, Q_2, \ldots, Q_k)$ be a k-partial Grundy witness. Suppose $Y_1, Y_2, \ldots, Y_k$ are pairwise disjoint independent sets in G such that $Q_i \subseteq Y_i$, for all $i \in [k]$. Then $(Y_1, Y_2, \ldots, Y_k)$ is also a k-partial Grundy witness of G.*

A k-partial Grundy witness $(X_1, X_2, \ldots, X_k)$ is *small* if for each $i \in [k]$, $|X_i| \leq k - i + 1$. Next, we prove the existence of a small k-partial Grundy witness. This result is the same as the one obtained by Effantin et al. [15]; however, it is stated slightly differently for convenience.

▶ **Observation 8 (♠).**[2] *Let G be a graph and k be an integer, where G has a k-partial Grundy witness $(Q_1, \dots, Q_k)$. Then, there exists a k-partial Grundy witness $(X_1, X_2, \dots, X_k)$ such that for each $i \in [k]$, $X_i \subseteq Q_i$ and $|X_i| \leq k - i + 1$.*

The remainder of this section is organized as follows. In Section 3.1 we prove our key structural result (Theorem 2), and then obtain our algorithm in Section 3.2. (Readers who may want to read the algorithm directly, may skip Section 3.1.)

3.1 Degree Reduction: Proof of Theorem 2

The objective of this section is to prove Theorem 2. The proof of this theorem is based on the following lemma for bipartite graphs.

▶ **Lemma 9.** *There is a polynomial-time algorithm that, given a bipartite graph $G = (A \uplus B, E)$ and a positive integer k, does one of the following.*

(i) *Correctly concludes that the partial Grundy coloring of G is at least k.*

(ii) *Outputs at most $4k - 4$ bicliques $A_1, \cdots, A_\ell$ in G such that for any $v \in V(G)$, degree of v in $G - F$ is at most k^2, where F is the union of the edges in the above bicliques.*

We first give a proof of Theorem 2 based on the above lemma.

Proof of Theorem 2. Consider a graph G and a positive integer k. First, we run the first-fit greedy algorithm for proper coloring of the graph G for an arbitrarily fixed ordering $(v_1, v_2, \cdots, v_n)$ of $V(G)$. For each j, let C_j be the vertices colored j and k' be the largest integer such that $C_{k'} \neq \emptyset$. Note that $(C_1, \cdots, C_{k'})$ is a proper coloring of G. Also, for any $j \in [k']$ and any vertex v in C_j, v has a neighbor in $C_{j'}$ for all $j' \in [j-1]$. If $k' \geq k$, then $(C_1, \cdots, C_{k'})$ is a partial Grundy coloring of G using at least k colors, and thus we can correctly report it.

Next, we assume that $k' < k$. Note that all the edges in G are between the color classes $C_1, \cdots, C_{k'}$. Now for every distinct $i, j \in [k']$, where $i < j$, we apply Lemma 9 on $(H_{i,j} = (C_i \uplus C_j, E(C_i, C_j)), k)$, where $E(C_i, C_j)$ is the set of edges in G between the color classes C_i and C_j. Our algorithm will declare that G has a partial Grundy coloring using at least k colors if we get the output given in statement (i) in any of the $\binom{k'}{2}$ applications of Lemma 9. Otherwise, for every distinct $i, j \in [k']$, where $i < j$, let $A_{i,j,1}, \cdots, A_{i,j,\ell_{i,j}}$ be the bicliques, we get as output by the algorithm in Lemma 9 on $(H_{i,j}\ , k)$. Note that $\ell_{i,j} \leq 4k - 4$. Now our algorithm will output the bicliques $\{A_{i,j,r} :\ 1 \leq i < j \leq k', r \in [\ell_{i,j}]\}$. As $k' < k$ for all $1 \leq i < j \leq k'$, the number of bicliques we output is at most $(4k-4)\binom{k'}{2}$, that is, at most $2k^3$. Since any vertex in a color class C_r has neighbors in other color classes, for $r \in [k']$ and we applied Lemma 9 for every pair of color classes, the degree of v in $G - F$ is at most k^3 for any $v \in V(G)$, where F is the union of the edges in the above bicliques. This completes the proof of the theorem. ◀

We now focus on the proof of Lemma 9. Toward this, we give a polynomial time procedure that, given a bipartite graph $G = (L \uplus R, E)$ and a positive integer k, either concludes that the input graph has partial Grundy coloring using at least k colors or it outputs at most $2k - 2$ bicliques $A_1, \dots, A_\ell$ in G such that for any $v \in L$, $d_{G-F}(v) \leq k^2$, where F is the union of the edges in the above bicliques. That is, removal of the edges of these bicliques bounds the degree of each vertex in L by k^2. We get the proof of Lemma 9 by applying this algorithm once for L and then for R.

[2] The proofs of the result marked with ♠ can be found in the full version of the paper on arXiv.

Overview of our algorithm. Let $\sigma = v_1, v_2, \ldots, v_n$ be an ordering of the vertices in L in non-increasing order of their degree in G. The algorithm constructs *specific* color classes $Q_1, Q_2, \ldots, Q_r$ in this order so that $(C_1 = Q_r, C_2 = Q_{r-1}, \ldots, C_r = Q_1)$ is an r-partial Grundy witness, where $|Q_j| \leq j$. Furthermore, we will construct sets B_i, $i \in [r]$, which will be used to construct the bicliques. Notice that if we obtain $r \geq k$, then we will be able to conclude that G has a partial Grundy coloring using at least k colors. Let $Q_1 = \{v_1\}$ and in our construction v_1 will be the dominator in Q_1 (and our construction needs to ensure this property; see Definition 6). Consider the construction of Q_2. Let i be the smallest index in $\{2, 3, \ldots, n\}$ for which there is a vertex $w \in N_G(v_1)$ such that v_i is not adjacent to w. Then, we set $Q_2 = \{v_i, w\}$, and designate v_i as the dominator in color class $C_{r-1} = Q_2$. Notice that all the vertices in $B_1 = \{v_2, \ldots, v_{i-1}\}$ are adjacent to all the vertices in $N_G(v_1)$, and hence they together form a biclique (with bipartition B_1 and $N_G(v_1)$). This property will be extended in building each Q_js and the required bicliques.

For the construction of Q_j, we consider *unprocessed vertices* (i.e., the vertices that do not belong to the previously constructed sets, i.e., to $Q_1 \cup B_1 \cup \ldots \cup Q_{j-1} \cup B_{j-1}$) as follows. We would now like to choose an unprocessed vertex $v_{i'}$, so that we can make $v_{i'}$ the dominator of Q_j, and additionally, for each $j' \in [j-1]$, we can include a neighbor of the dominator from $Q_{j'}$ to the set Q_j. Note for us to do the above, we need to ensure that the vertices that we add to Q_j is an independent set in G, and all the vertices that we want to include in the set Q_j are outside $Q_1 \cup \ldots \cup Q_{j-1}$. That is, among the unprocessed vertices, we choose the first vertex $v_{i'}$ with the following property: for each $j' \in [j-1]$, we have a neighbor $w_{j'}$ of the previously constructed dominator in $Q_{j'}$ such that $w_{j'} \notin Q_1 \cup \ldots \cup Q_{j-1}$ and $(v_{i'}, w_{j'}) \notin E(G)$; we set $Q_j = \{v_{i'}, w_1, \ldots, w_{j-1}\}$. We would like to mention that all the dominators we construct are from the bipartition L and hence $\{w_1, \ldots, w_{j-1}\} \subseteq R$. This will imply that Q_j is an independent set.

Moreover, by the choice of $v_{i'}$ as the smallest vertex with the desired property, it follows that for any vertex v_r that appears before $v_{i'}$ in the order σ and $v_r \notin P = Q_1 \cup \ldots \cup Q_{j-1}$, the vertex v_r is adjacent to all the vertices in $N(x_{j'}) \setminus P$, where $x_{j'}$ is the dominator in $Q_{j'}$, for some $j' \in [j-1]$. Then we add v_r to $B_{j'}$. Note that in the above process, we still maintain the biclique property, by explicitly ensuing that $B_{j'}$ and $N(x_{j'}) \setminus (Q_1 \cup \ldots \cup Q_{j-1} \cup Q_j)$ forms a biclique.

Description of the algorithm. We give a pseudocode of our algorithm in Algorithm 1. First, the algorithm intializes the sets B_i and Q_i to be the empty set, for all $i \in [k]$ (see Algorithm 1). Let $\sigma = v_1, v_2, \ldots, v_n$ be an ordering of the vertex set L in the non-increasing order of their degrees. Now, we want to construct the color classes $Q_1, Q_2, \ldots, Q_k$, iteratively, such that $(C_1, C_2, \ldots, C_k) = (Q_k, Q_{k-1}, \ldots, Q_1)$ is a k-partial Grundy witness. At line 3, we intialize with $Q_1 := \{v_1\}$, $x_1 := v_1$, and fix the index $q = 2$. Here, Q_1 will be the color class C_k with dominator vertex x_1. Now consider an iteration of the **while** loop. The algorithm checks if the set $L \setminus \left(\bigcup_{j \in [q-1]} (Q_j \cup B_j)\right)$ is non-empty and executes the **while** loop. At this point we have constructed sets $Q_1, \ldots, Q_{q-1}$ such that $(Q_{q-1}, Q_{q-2}, \ldots, Q_1)$ is a $(q-1)$-partial Grundy witness such that each x_i is a dominator vertex in Q_i. Let v_r be the first unprocessed vertex in L and $P = \cup_{j \in [q-1]} Q_j$ by Lines 5 and 6. Now, we check if we can construct the current color class Q_q with vertex v_r as a dominator vertex, and for that, we need to add a neighbor w_j (which is not added to any $Q_{i'}$ before) for each already discovered dominator x_j such that w_j is non-adjacent to v_r. Now, if there exists some $j \in [q-1]$ such that each neighbor of x_j is a neighbor of v_r , then we will not be able to construct Q_q with vertex v_r in it. In that case, we choose such a value j and add v_r to B_j(See

Algorithm 1 Algorithm for one-sided bipartite structural result.

1 Initialize $B_i := \emptyset$ and $Q_i := \emptyset$, for each $i \in [k]$.
2 Let $\sigma = v_1, \cdots, v_n$ be an order of the vertices in L in the non-increasing order of their degrees.
3 Let $x_1 := v_1$, $Q_1 := \{x_1\}$ and $q := 2$.
4 **while** $L \not\subseteq \bigcup_{j \in [q-1]} (Q_j \cup B_j)$ **do**
5   Let v_r be the first vertex in σ from $L \setminus \left(\bigcup_{j \in [q-1]} (Q_j \cup B_j) \right)$.
6   Let $P = \bigcup_{j \in [q-1]} Q_j$.
7   **if** *there exists* $j \in [q-1]$ *such that* $N(x_j) \subseteq P \cup N(v_r)$ **then**
8   choose an arbitrary j with this property and set $B_j := B_j \cup \{v_r\}$.
9   **else**
10   For each $j \in [q-1]$, $N(x_j) \setminus (P \cup N(v_r)) \neq \emptyset$. Then, for each $j \in [q-1]$ arbitrarily pick a vertex w_j from $N(x_j) \setminus (P \cup N(v_r))$.
11   //(Notice that in the above, w_j *may be equal to* $w_{j'}$ for distinct $j, j' \in [q-1]$).
12   Set $x_q := v_r$.
13   Set $Q_q := \{x_q\} \cup \{w_1, \cdots, w_{q-1}\}$.
14   Set $q := q + 1$.
15   **end**
16 **end**
17 **if** $q \geq k + 1$ **then**
18   Declare that partial Grundy coloring of G is at least k.
19 **else**
20   For all $j \in [q-1]$, let A_j be the bipartite graph induced on B_j union $N(x_j) \setminus P$ and S_j be the graph induced on $N[x_j]$, where $P = \bigcup_{j \in [q-1]} Q_j$.
21   Output $A_1, \cdots, A_{q-1}$ and $S_1, \cdots, S_{q-1}$
22 **end**

Line 8). Here, notice that v_r is adjacent to all the vertices $N(x_j) \setminus P$. We will maintain this property for all the vertices added to B_j, i.e., B_j union $N(x_j) \setminus \bigcup_i Q_i$ forms a biclique. Now consider the case that the condition in the **if** statement in Line 7 is false. Then, we choose a vertex $w_j \in N(x_j) \setminus (P \cup N(v_r))$ for each $j \in [q-1]$, by line 11 and set the vertex v_r as the dominator for Q_q, that is, $Q_q := \{x_q\} \cup \{w_1, \cdots, w_{q-1}\}$, at Line 13. Notice that Q_q is an independent set because there is no edge between x_1 and a vertex in $\{w_1, \ldots, w_{q-1}\}$, and $\{w_1, \ldots, w_{q-1}\}$ is a subset of R, the right part of the bipartition of G. We repeat the iteration until one of the **while** loop conditions at line 4 fails. Next, if $q \geq k+1$, we conclude that G has a partial Grundy coloring using k colors by line 18, because $(Q_{q-1}, \ldots, Q_1)$ is a $(q-1)$-partial Grundy witness, where $q - 1 \geq k$. Otherwise, by line 20, let A_j be the graph induced on $B_j \cup (N(x_j) \setminus P)$ and S_j be the graph induced on $N[x_j]$, for each $j \in [q-1]$. Recall that A_j is a biclique. It is easy to see that S_j is a biclique, because G is a bipartite graph. At line 21, the algorithm outputs the set of graphs $A_1, \cdots, A_{q-1}$ and $S_1, \cdots, S_{q-1}$.

The number of iterations of the **while** loop is at most n and each step in the algorithm takes polynomial time, the total running time of the algorithm is polynomial in the input size. Next, we prove the correctness of the algorithm.

▶ **Lemma 10.** *Algorithm 1 is correct.*

Proof. Let $q^\star$ be the value of q at the end of the algorithm. To prove the correctness of the algorithm, first, we prove the following claim.

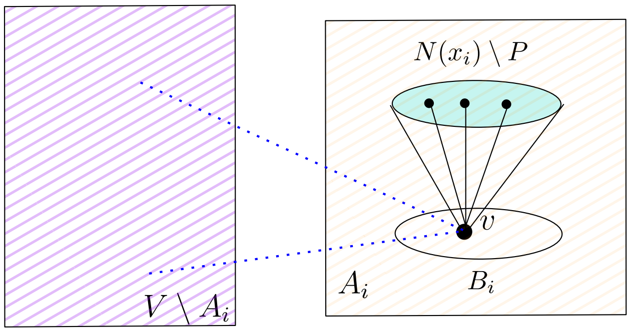

Figure 2 Here the vertex $v \in B_i$ in the biclique A_i (right side). The number of neighbors of v outside A_i (blue edges) cannot be more than $|P|$ as $d_G(v) \leq d_G(x_i) = |N_G(x_i)|$ and v has $|N(x) \setminus P|$ neighbours in the biclique A_i.

▷ Claim 11. The following statements are true.

(i) For each $i \in [q^\star - 1]$, Q_i is an independent set and $Q_i \neq \emptyset$.

(ii) For each $i \in [q^\star - 1]$ and $j \in [i-1]$, $N(x_j) \cap Q_i \neq \emptyset$.

(iii) For each $i \in [q^\star - 1]$ and $v \in B_i$, v is adjacent to all the vertices in $N(x_i) \setminus (\bigcup_{j \in [q^\star - 1]} Q_j)$ and $d_G(v) \leq d_G(x_i)$.

Proof. We prove the statements by induction on i. The base case is when $i = 1$. Clearly, $Q_1 = \{x_1\}$ and hence statement (i) is true. Statement (ii) is vacuously true. Next, we prove statement (iii). Notice that in any iteration of the **while** loop, in Step 8, we may add a vertex v_r to B_1. If this happens, then we know that $N(x_1) \subseteq P \cup N(v_r)$, where P is a subset of $\bigcup_{j \in [q^\star - 1]} Q_j$. That is, all the vertices in $N(x_1) \setminus P$ are adjacent to v_r. This implies that v_r is adjacent to all the vertices in $N(x_1) \setminus (\bigcup_{j \in [q^\star - 1]} Q_j)$. Since x_1 is the vertex with the maximum degree, we have that $d_G(v_r) \leq d(x_1)$.

Next, for the induction step, we assume that the induction hypothesis is true for $i-1$, and we will prove that the hypothesis is true for i. Consider the iteration $h^\star$ of the **while** loop when $q = i$ and Steps 11-14 is executed. Let v_r be the vertex mentioned in Step 5 during that iteration. The vertices $w_1, \cdots, w_{q-1}$ belongs to R (the right side of the bipartition of G) and hence $\{w_1, \cdots, w_{q-1}\}$ is an independent set. Also, notice that each w_j does not belong to $N(v_r)$ (See Step 11). Hence, $Q_q := \{v_r\} \cup \{w_1, \cdots, w_{q-1}\}$ is an independent set and $Q_q \neq \emptyset$. Thus, we proved statement (i). Again, notice that $w_j \in N(x_j)$ for all $j \in [q-1]$ (See Step 11). Thus, statement (ii) follows. Next, we prove statement (iii), which is similar to the proof of it in the base case. Notice that in any iteration of the while loop (after the iteration $h^\star$), in Step 8, we may add a vertex $v_{r'}$ to B_i. If this happens, then we know that $N(x_i) \subseteq P \cup N(v_{r'})$, where P is a subset of $\bigcup_{j \in [q-1]} Q_j$. That is, all the vertices in $N(x_i) \setminus P$ are adjacent to $v_{r'}$. This implies that $v_{r'}$ is adjacent to all the vertices in $N(x_i) \setminus (\bigcup_{j \in [q-1]} Q_j)$. Since $x_i \in Q_i$, considered in iteration $h^\star$, $d_G(v_{r'}) \leq d_G(x_i)$ (See Step 5). This completes the proof of the claim. ◁

Now suppose $q^\star \geq k+1$. Then, by Statements (i) and (ii) in Claim 11, we get that $(Q_k, \cdots, Q_1)$ is a partial Grundy coloring of the graph induced on $\bigcup_{j \in [q^\star - 1]} Q_j$. Thus, if the algorithm executes Step 18, then it is correct because of Observation 5.

Now suppose $q^\star \leq k$. Then the algorithm executes Step 21 and outputs the sets $A_1, \ldots, A_{q^\star - 1}$ and $S_1, \ldots, S_{q^\star - 1}$. Statement (iii) in Claim 11 implies that each A_j is a biclique in G, where $j \in [q^\star - 1]$. Also, note that S_j is a biclique in G as it is induced on the set $N[x_j]$, for each $j \in [q^\star - 1]$. Let F be the union of the edges in the bicliques

$A_1, \ldots, A_{q^\star-1}$ and $S_1, \ldots, S_{q^\star-1}$. Next, we prove that for any $v \in L$, $d_{G-F}(v) \leq k^2$. Let $X = \{x_1, \ldots, x_{q^\star-1}\}$. It is easy to see that $|Q_j| \leq k$ and $L \cap Q_j = \{x_j\} \subseteq X$, for all $j \in [q^\star - 1]$ (See Steps 11-13). Hence, $|\bigcup_{j \in [q^\star]} Q_j| \leq k^2$. Let v be an arbitrary vertex in L. Note that if $v \in X$, the degree of v in $G - F$ is zero (by the definition of S_j). Next, suppose that $v \in L \setminus X$. Since $L \subseteq \bigcup_{j \in [q^\star-1]} (Q_j \cup B_j)$ (this is the condition for **while** loop to exit) and $L \cap Q_j \subseteq X$ for all $j \in [q^\star - 1]$, v belongs to B_i for some $i \in [q^\star - 1]$. By statement (iii) in Claim 11, v is adjacent to all the vertices in $N(x_i) \setminus (\bigcup_{j \in [q^\star-1]} Q_j)$ and $d_G(v) \leq d_G(x_i)$. Recall that the biclique A_i is the graph with bipartition B_i and $N(x_i) \setminus (\bigcup_{j \in [q^\star-1]} Q_j)$. Moreover $v \in B_i$ and $d_G(v) \leq d_G(x_i) = |N_G(x_i)|$. This implies that the number of neighbors of v that does not belong to A_i is at most $|\bigcup_{j \in [q^\star-1]} Q_j|$, which is upper bounded by k^2. Therefore, the degree of v in $G - F$ is at most k^2. See Figure 2 for an illustration. This concludes the proof. ◀

3.2 FPT Algorithm for Partial Grundy Coloring

In this section, we design an FPT algorithm $\mathcal{A}$ for Partial Grundy Coloring when the input has a structure dictated the second item of Lemma 9. Then, we explain how to get an FPT algorithm for Partial Grundy Coloring on general graphs using $\mathcal{A}$ and Theorem 2. Moreover, our algorithm $\mathcal{A}$ will provide a faster algorithm for Partial Grundy Coloring on d-degenerate graphs, improving the result in [1]. Toward the first task, we define the following problem.

Structural Partial Grundy Coloring (SPGC)
Input: Positive integers $k, d, \ell \in \mathbb{N}$, a graph G, and ℓ bicliques $A_1, \ldots, A_\ell$ in G such that $G - F$ is d-degenerate, where F is the union of the edges in the above bicliques.
Question: Decide if there is a partial Grundy coloring for G using at least k colors.

First, we design a randomized polynomial time algorithm $\mathcal{A}_1$ for SPGC with a success probability at least $(k(d+1))^{-2k^2-k} \cdot 2^{-\ell k}$. We increase the probability of success to a constant by running $\mathcal{A}_1$ multiple times. Finally, we explain the derandomization of our algorithm. Then we prove Theorem 1 using this algorithm and our structural result (Theorem 2). To design the algorithm $\mathcal{A}_1$, we use the following result of Lokshtanov et al. [28].

▶ **Proposition 12** (Lemma 1.1. [28]). *There is a linear-time randomized algorithm that, given a d-degenerate graph H and an integer k, outputs an independent set Y such that for any independent set X in H with $|X| \leq k$, the probability that $X \subseteq Y$ is at least* $\left(\binom{k(d+1)}{k} \cdot k(d+1)\right)^{-1}$.

The algorithm $\mathcal{A}_1$ has the following steps.

1. Color all vertices in $V(G)$ uniformly and independently at random with colors from the set $[k]$. Let the obtained coloring be $\phi : V(G) \to [k]$, and $Z_i = \phi^{-1}(i)$, for each $i \in [k]$.
2. For each $i \in [\ell]$, let $A_i = (L_i \uplus R_i, E_i)$. For each $j \in [k]$ and $i \in [\ell]$, uniformly at randomly assign $P_{j,i} := L_i$ or $P_i := R_i$. That is, with probability $\frac{1}{2}$, $P_{j,i} := L_i$ and with probability $\frac{1}{2}$, $P_{j,i} := R_i$. Let $D_j = \bigcup_{i \in [\ell]} P_{j,i}$.
3. Now for each $j \in [k]$, we apply the algorithm in Proposition 12 for $(G[Z_j - D_j], k)$ to obtain an independent set Y_j.
4. If $(Y_1, \ldots, Y_k)$ is a k-partial Grundy witness of G, then output **Yes**, else, output **No**.

Since the algorithm in Proposition 12 runs in linear time, the algorithm $\mathcal{A}_1$ can be implemented to run in linear time because $Z_1, \ldots, Z_k$ is a partition of $V(G)$. Clearly, if the algorithm $\mathcal{A}_1$ outputs **Yes**, then G has a k-Partial Grundy witness and hence G has a partial Grundy coloring using at least k colors. We can prove that if G has a k-Partial Grundy witness the algorithm $\mathcal{A}_1$ outputs **Yes** with probability $(k(d+1))^{-2k^2-k} \cdot 2^{-\ell k}$.

Also, by running $\mathcal{A}_1$, $3 \cdot (k(d+1))^{2k^2+k} 2^{\ell k}$ times and outputting **Yes** if at least one of the runs outputs a **Yes**, and outputs **No**, otherwise, we can boost the success probability to $2/3$, and thus obtain the following result.

▶ **Theorem 13.** *There is a randomized algorithm for* SPGC *running in time $\mathcal{O}((k(d+1))^{2k^2+k} \cdot 2^{\ell k} \cdot (m+n))$. In particular, if (G,k) is a no-instance then with probability* 1 *the algorithm outputs* **No***; and if (G,k) is a yes-instance then with probability* 2/3 *the algorithm outputs* **Yes***.*

Theorem 2 and 13 imply the following theorem.

▶ **Theorem 14.** *There is a randomized algorithm for* Partial Grundy Coloring *running in time $2^{\mathcal{O}(k^4)} n^{\mathcal{O}(1)}$. In particular, if (G,k) is a no-instance then with probability* 1 *the algorithm outputs* **No***; and if (G,k) is a yes-instance then with probability* 2/3 *the algorithm outputs* **Yes***.*

Proof Sketch. First we run the algorithm mentioned in Theorem 2. If it concludes that G has a partial Grundy coloring with at least k colors, then we output **Yes**. Otherwise, we get at most $2k^3$ induced bicliques $A_1 \cdots, A_\ell$ in G such that the following holds. For any $v \in V(G)$, the degree of v in $G - F$ is at most k^3, where F is the union of the edges in the above bicliques. That is the degeneracy of $G - F$ is at most k^3. Then, we apply Theorem 13, and ouputs accordingly. ◀

The derandomization of our algorithm can be found in the full version.

4 FPT Algorithm for Grundy Coloring on $K_{i,j}$-free Graphs

This section aims to prove Theorem 3. Consider fixed $i, j \in \mathbb{N} \setminus \{0\}$, where $i \geq j$. Recall that a graph is $K_{i,j}$-free if it does not contain a subgraph isomorphic to $K_{i,j}$. We call the special case of Grundy Coloring where the input graph is $K_{i,j}$-free, $K_{i,j}$-Free Grundy Coloring. Let (G,k) be an instance of $K_{i,j}$-Free Grundy Coloring. We begin by intuitively explaining the flow of our algorithm.

Consider a Grundy coloring $c : V(G) \to [k']$ of G, where $k' \geq k$, and for each $z \in [k]$, $c^{-1}(z) \neq \emptyset$. Furthermore, for $z \in [k']$, let $C_z = c^{-1}(z)$. Let us focus on the first k color classes, and for $z \in [k]$, arbitrarily fix a vertex $v_z \in C_z$. (Note that v_z has a neighbor in $C_{z'}$, for each $z' \in [z-1]$.) We next intuitively describe construction, for each $z \in [k]$, a set W_z initialized to $\{v_z\}$ as follows. Basically, for each v_z, add an arbitrarily chosen neighbor of it in color class $C_{z'}$, for every $z' < z$. We do the above process exhaustively; whenever we add a vertex to a set W_z, we add an arbitrarily chosen neighbor of it from each color class $C_{z'}$ to $W_{z'}$, where $z' < z$. Then, let $W = \cup_{z \in [k]} W_z$; we will call such a set W a *k-Grundy set* for G and we will show that such a set of size at most 2^{k-1} exists (for yes instances). For $z \in [k]$, let $W_{\leq z} = \cup_{z' \in [z]} W_{z'}$ and $W_{>z} = \cup_{z' \in [k] \setminus [z]} W_{z'}$. Note that $c|_W$ is a k-Grundy coloring of $G[W]$. Also, we will show that G has a Grundy coloring using at least k colors if and only if some subgraph of G has a Grundy coloring using *exactly* k colors. We remark that the above result and the existence of W are the same as the results of Gyárfás et al. [22] and Zaker [38], although, for the sake of convenience, we state it here slightly differently. If we

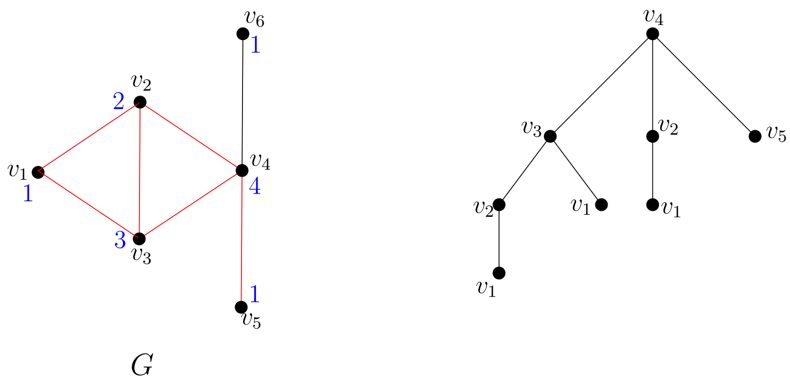

Figure 3 An illustration of a graph G that admits a 4-Grundy coloring (on the left) and a 4-Grundy-witness ω (on the right).

can identify all the vertices in W (or some other k-Grundy set), then we will be done. The first step of our algorithm will be to use the technique of color coding to randomly color the vertices of G using k colors so that, for each $z \in [k]$, $v \in W_z$ is colored z; such a coloring will be a *nice coloring* and it will be denoted by χ.

The next step of our algorithm is inspired by the design of FPT algorithms based on computations of "representative sets" [19,30]. To this end, we will interpret W in a "tree-like" fashion. With this interpretation, in a bottom-up fashion, for each $z \in [k]$ and $v \in X_z$, we will compute a family $\mathcal{F}'_{z,v}$, so that, if $v \in W_z$, then there will be $W' \in \mathcal{F}'_{z,v}$ so that $W' \cup W_{>z}$ is also a k-Grundy set for G. We will now formalize the above steps.

Grundy Tree & Grundy Witness. We recall the Definition 4 from Section 1, and obtain some properties regarding it.

▶ **Observation 15 (♠).** *For $k \in \mathbb{N} \setminus \{0\}$, for the k-Grundy tree (T_k, ℓ_k), $|V(T_k)| = 2^{k-1}$.*

▶ **Observation 16 (♠).** *Consider $k \in \mathbb{N} \setminus \{0\}$ and the k-Grundy tree (T_k, ℓ_k). We have $|\ell^{-1}(k)| = 1$ and for each $z \in [k-1]$, $|\ell_k^{-1}(z)| = 2^{k-z-1}$.*

Next, we define the notion of *k-Grundy witness* in a graph G.

▶ **Definition 17.** Consider $k \in \mathbb{N}\setminus\{0\}$ and a graph G. A *k-Grundy witness* for G is a function $\omega : V(T_k) \to V(G)$, where (T_k, ℓ_k) is the k-Grundy tree, such that: 1) for each $z \in [k]$, $\{\omega(t) \mid t \in V(T_k)$ and $\ell_k(t) = z\}$ is an independent set in G, 2) for each $t, t' \in V(T_k)$, if $\ell_k(t) \neq \ell_k(t')$ then $\omega(t) \neq \omega(t')$, and 3) for each $(t, t') \in A(T_k)$, we have $\{\omega(t), \omega(t')\} \in E(G)$.

Recall that for $k \in \mathbb{N}\setminus\{0\}$, for the k-Grundy tree (T_k, ℓ_k), T_k is the tree obtained by adding a root vertex r_k attached to the roots of (pairwise vertex disjoint) trees $T_{k-1}, T_{k-2}, \cdots, T_1$, where for each $z \in [k-1]$, (T_z, ℓ_z) is the z-Grundy tree. We have the following observation.

▶ **Observation 18 (♠).** *Consider $k \in \mathbb{N} \setminus \{0\}$, a graph G and a k-Grundy witness $\omega : V(T_k) \to V(G)$ for G. For each $z \in [k]$, $\omega|_{V(T_z)}$ is a z-Grundy witness for G.*

The next observation is a partial Grundy counterpart of Observation 5.

▶ **Observation 19 (♠).** *Consider a graph G, any induced subgraph $\widehat{G}$ of it, and an integer $k \in \mathbb{N}$. If $\widehat{G}$ has a Grundy coloring that uses exactly k colors, then G has a Grundy coloring that uses at least k colors.*

In the following two lemmas, we show that the existence of a k-Grundy witness for a graph is equivalent to the graph admitting a Grundy coloring with at least k colors.

▶ **Lemma 20.** *For any $k \in \mathbb{N} \setminus \{0\}$ and a graph G, if G has a k-Grundy witness, then G has a Grundy coloring with at least k colors.*

Proof. Consider a graph G and any $k \in \mathbb{N}\setminus\{0\}$. For a k-Grundy witness $\omega : V(T_k) \to V(G)$ of G, let $\widehat{V}_\omega = \{\omega(t) \mid t \in V(T_k)\}$, and for each $z \in [k]$, let $\widehat{V}_{\omega,z} = \{\omega(t) \mid t \in V(T_k) \text{ and } \ell_k(t) = z\}$. Note that from item 2 of Definition 17, $\widehat{V}_{\omega,1}, \widehat{V}_{\omega,2}, \cdots, \widehat{V}_{\omega,k}$ is a partition of $\widehat{V}_\omega$, where none of the parts are empty. Let $c_\omega : \widehat{V}_\omega \to [k]$ be the function such that for each $z \in [k]$ and $v \in \widehat{V}_{\omega,z}$, we have $c_\omega(v) = q$.

For each $k \in \mathbb{N} \setminus \{0\}$ and a k-Grundy witness $\omega : V(T_k) \to V(G)$ of G, we will prove by induction (on k) that c_ω is Grundy coloring of $G[\widehat{V}_\omega]$ using k colors. The above statement, together with Observation 19, will give us the desired result.

The base case is $k = 1$, where T_1 has exactly one vertex, r_1. For any 1-Grundy witness, ω of G, note that $c_\omega(\omega(r_1)) = 1$ is a Grundy coloring for $G[\{\omega(r_1)\}]$ using 1 color. Now for the induction hypothesis suppose that for some $\widehat{k} \in \mathbb{N} \setminus \{0, 1\}$, for each $0 < k < \widehat{k}$, the statement is true. Now we will prove the statement for $k = \widehat{k}$, and to this end, we consider a k-Grundy witness $\omega : V(T_k) \to [k]$, where r_k is the root of T_k. Recall that T_k is the tree obtained by adding a root vertex r_k attached to the roots of (pairwise vertex disjoint) trees $T_{k-1}, T_{k-2}, \cdots, T_1$, where for each $z \in [k-1]$, (T_z, ℓ_z) is the z-Grundy tree, and T_z is rooted at r_z. Let $V' = \widehat{V}_\omega \setminus \widehat{V}_{\omega,k}$, and consider a vertex $v \in \widehat{V}_{\omega,z^*}$, where $z^* \in [k-1]$. We will argue that for each $z' \in [z^* - 1]$, $N_G(v) \cap \widehat{V}_{\omega,z'} \neq \emptyset$. Note that there must exists $z \in [k-1]$ and $t \in V(T_z)$ such that $\omega(t) = v$ and $\ell_z(v) = z^*$, and we arbitrarily choose one such z and t. Let $V_z = \{\omega(t) \mid t \in V(T_z)\}$. From Observation 18, $\omega_z = \omega|_{V(T_z)}$ is a z-Grundy witness for G. Thus, from our induction hypothesis, $c_{\omega_z} = c_\omega|_{V_z}$ is a Grundy coloring for $G[V_z]$. From the above we can conclude that for each $q' \in [z^* - 1]$, $N_G(v) \cap \widehat{V}_{\omega,z'} \neq \emptyset$. Now consider the vertex $\omega(r_k) = v^*_k$ and any $z \in [k-1]$. Note that $\ell_k(r_z) = z$ and from item 3 of Definition 17, we can obtain that $\{v^*_k, \omega(r_z)\} \in E(G)$. From the above discussions, we can obtain that c_ω is Grundy coloring of $G[\widehat{V}_\omega]$ using at least z colors. This concludes the proof. ◀

▶ **Lemma 21.** *For any $k \in \mathbb{N} \setminus \{0\}$ and a graph G, if G has a Grundy coloring with at least k colors, then G has a k-Grundy witness.*

Proof. Consider a Grundy coloring $c : V(G) \to [k']$ of G with $k' \geq k$ colors, and for each $q \in [k']$, let $C_q = c^{-1}(q)$. We construct a Grundy witness $\omega : V(T_k) \to V(G)$ by processing labels of T_k starting at k and iteratively proceeding to smaller labels as follows while maintaining the below invariants.
Pre-condition: When we begin processing a label $q \in [k-1]$, for each $t \in V(T_k)$ with $\ell_k(t) \geq q$, we have fixed the vertex $\omega(t)$.
Post-condition: After processing label $q \in [k]$, we have fixed, for each $t \in V(T_k)$ with $\ell_k(t) \geq q$, and $t' \in N_{T_k}[t]$, the vertex $\omega(t')$; and these are the only vertices in T_k for which the vertex in G assigned by ω is determined.

Note that the pre-condition is vacuously satisfied for $q = k$. Recall that T_k is the tree obtained by adding a root vertex r_k attached to the roots $r_{k-1}, r_{k-2}, \cdots, r_1$ of (pairwise vertex disjoint) trees $T_{k-1}, T_{k-2}, \cdots, T_1$, respectively, where for each $q \in [k-1]$, (T_q, ℓ_q) is a q-Grundy tree. Pick any vertex $v_k \in C_k$, and set $\omega(r_k) := v_k$ and for each $q \in [k-1]$, set $\omega(r_q) := w^k_q$, where w^k_q is an arbitrarily chosen neighbor of v_k from C_q (which exists as c is a Grundy coloring). After the above step, the post-condition is satisfied for $q = k$.

Now we (iteratively, in decreasing order) consider $q \in [k-1] \setminus \{1\}$. From the pre-condition for q, we have fixed, for each $t \in V(T_k)$ with $\ell_k(t) \geq q$, the vertex $\omega(t)$. Consider $t \in V(T_k)$ with $\ell_k(t) = q$ and let $v_t = \omega(t)$. Let $\widehat{T}_q$ be the subtree of T_k rooted at t, and let

$\widehat{\ell}_q = \ell_k|_{V(\widehat{T}_q)}$. Notice that $(\widehat{T}_q, \widehat{\ell}_q)$ is a q-Grundy tree, where $\widehat{T}_q$ is the tree obtained from by adding edge between t and the roots $\widehat{r}_{q-1}, \widehat{r}_{q-2}, \cdots, \widehat{r}_1$ of $\widehat{T}_{q-1}, \widehat{T}_{q-2}, \cdots, \widehat{T}_1$, respectively, where $(\widehat{T}_{q'}, \ell_k|_{V(\widehat{T}_{q'})})$ is a q'-Grundy tree, for each $q' \in [q-1]$. For each $q' \in [q]$, let $\widehat{w}^q_{q'}$ be an arbitrarily chosen vertex from $N_G(v) \cap C_{q'}$, and we set $\omega(\widehat{r}_{q'}) = \widehat{w}^q_{q'}$. Notice that after the above step, the post-condition is satisfied for q, and the pre-condition is satisfied for $q-1$.

After we are done processing each $q \in [k] \setminus \{1\}$, the post-condition for $q = 2$ (and the pre-condition of $q = 1$ implies that for each $t \in V(T_k)$, we have determined the vertex $\omega(t)$). Moreover, the construction of ω implies that all the three conditions in Definition 17 are satisfied. This concludes the proof. ◀

We next summarize the result we obtain from the above two lemmas.

▶ **Lemma 22.** *Consider any $k \in \mathbb{N} \setminus \{0\}$ and a graph G. The graph G has a k-Grundy witness if and only if G has a Grundy coloring with at least k colors.*

Color Coding of G. We will next use the above lemma to simplify our job in the following sense. Let $\omega : V(T_k) \to V(G)$ be a (fixed) k-Grundy witness of G (if it exists), where (T_k, k) is a k-Grundy tree. Let $\widehat{V}_\omega = \{\omega(t) \mid t \in V(T_k)\}$, and for each $q \in [k]$, let $\widehat{V}_{\omega,q} = \{\omega(t) \mid t \in V(T_k)$ and $\ell_k(t) = q\}$. Roughly speaking, our new objective will be to find the vertices in $\widehat{V}_\omega$ and say that $G[\widehat{V}_\omega]$ admits a Grundy coloring with at least k colors, using which we can conclude that G admits a Grundy coloring with at least k colors. We will use the technique of color coding introduced by Alon et al. [2], to color the vertices in $\widehat{V}_\omega$ "nicely" as follows. Color each vertex in G uniformly at random using a color from $[k]$, and let $\chi : V(G) \to [k]$ be this coloring. A *nice* coloring is the one where, for each $q \in [k]$, the coloring assigns the color q to all the vertices in $\widehat{V}_{\omega,q}$.

We will work with the assumption that χ is a nice coloring of G, and for each $q \in [k]$, let $X_q = \chi^{-1}(q)$. Our objective will be to look for a k-Grundy witness $\widehat{\omega} : V(T_k) \to V(G)$, where (T_k, k) is a k-Grundy tree, such that for each $q \in [k]$ and $t \in V(T_k)$ with $\ell_k(t) = q$, we have $\widehat{\omega}(t) \in X_q$. To this end, we will store a "Grundy representative family" for each vertex in a bottom-up fashion, starting from $q = 1$. The definition of such a representative is inspired by the q-representative families [19, 30], although here we need a "vectorial" form of representation. To this end, we introduce the following notations and definitions.

Grundy Representative Sets. Recall we have the coloring χ of G with color classes $X_z = \chi^{-1}(z)$, for $z \in [k]$. A vertex subset $A \subseteq V(G)$ is *χ-independent* if for each $z \in [k]$, $A \cap X_z$ is an independent set in G. For $p \in \mathbb{N}$, a family of vertex subsets $\mathcal{F}$ is a *p-family* if each set in $\mathcal{F}$ has size at most p and each $A \in \mathcal{F}$ if χ-independent. We will only be working with vectors all of whose entries are from $\mathbb{N}$ without explicitly stating it. For a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$, $\mathsf{sum}(\overrightarrow{q})$ denotes the number $\sum_{z \in [q]} q_z$. For a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$ and $B \subseteq V(G)$, we say that the *size* of B is $\overrightarrow{q}$, written as $|B| = \overrightarrow{q}$, if for each $z \in [k]$, $|B \cap X_z| = q_z$. For vertex subsets A and B, A *fits* B if $A \cup B$ is χ-independent. For two vectors $\overrightarrow{q_1} = (q^1_1, q^1_2, \cdots, q^1_k)$ and $\overrightarrow{q_2} = (q^2_1, q^2_2, \cdots, q^2_k)$, and $\diamond \in \{\leq, \geq, >, <, =\}$, we write $\overrightarrow{q_1} \diamond \overrightarrow{q_2}$ if for each $z \in [k]$, we have $q^1_z \diamond q^2_z$. We next define the notion of $\overrightarrow{q}$-Grundy representation.

▶ **Definition 23.** Consider $p \in \mathbb{N}$, a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$, and a p-family $\mathcal{F}$ of vertex subsets of G. For a sub-family $\mathcal{F}' \subseteq \mathcal{F}$, we say that $\mathcal{F}'$ *$\overrightarrow{q}$-Grundy represents* $\mathcal{F}$, written as $\mathcal{F}' \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}$, if the following holds. For any set B of size $\overrightarrow{q}$, if there is $A \in \mathcal{F}$ that fits B, then there is $A' \in \mathcal{F}'$ that fits B. In the above, $\mathcal{F}'$ is a *$\overrightarrow{q}$-Grundy representative* for $\mathcal{F}$.

Next, we obtain some properties regarding $\overrightarrow{q}$-Grundy representatives.

▶ **Observation 24** (♠). *Consider $p \in \mathbb{N}$, a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$, and any two p-families $\mathcal{F}_1$ and $\mathcal{F}_2$. If $\mathcal{F}'_1 \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}_1$ and $\mathcal{F}'_2 \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}_2$, then $\mathcal{F}'_1 \cup \mathcal{F}'_2 \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}_1 \cup \mathcal{F}_2$.*

Consider $p \in \mathbb{N}$ and $v \in V(G)$. For a family $\mathcal{F}$ over $V(G)$, $\mathcal{F} + v$ denotes the family $\{A \cup \{v\} \mid A \in \mathcal{F} \text{ and } A \cup \{v\} \text{ is } \chi\text{-independent}\}$. Similarly, $\mathcal{F} - v$ denotes the family $\{A \setminus \{v\} \mid A \in \mathcal{F}\}$. A p-family $\mathcal{F}$ is a *(p, v)-family* if for each $A \in \mathcal{F}$, we have $v \in A$.

▶ **Observation 25** (♠). *Consider $p \in \mathbb{N}$, a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$, a vertex $v \in V(G)$ and a (p, v)-family $\mathcal{F}$. Let $\overrightarrow{h}$ be the vector obtained from $\overrightarrow{q}$ by increasing its $\chi(v)$th coordinate by 1. If $\mathcal{F}' \subseteq^{\overrightarrow{h}}_{grep} \mathcal{F} - v$ and $\mathcal{F}'' \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F} - v$, then $(\mathcal{F}' + v) \cup (\mathcal{F}'' + v) \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}$.*

For a p_1-family $\mathcal{F}_1$ and a p_2-family $\mathcal{F}_2$, we define a $(p_1 + p_2)$-family, $\mathcal{F}_1 \star \mathcal{F}_2 = \{A_1 \cup A_2 \mid A_1 \in \mathcal{F}_1, A_2 \in \mathcal{F}_2, \text{ and } A_1 \cup A_2 \text{ is } \chi\text{-independent}\}$. The following lemma will be helpful in obtaining a $\overrightarrow{q}$-representative for $\mathcal{F}_1 \star \mathcal{F}_2$.

▶ **Lemma 26.** *Consider a p_1-family $\mathcal{F}_1$, a p_2-family $\mathcal{F}_2$, and a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$, where $\mathsf{sum}(\overrightarrow{q}) + p_1 + p_2 \leq 2^{k-1}$. Let $\mathcal{F}'_1 \subseteq \mathcal{F}_1$ be a p_1-family such that for every vector $\overrightarrow{q_1} \geq \overrightarrow{q}$ with $\mathsf{sum}(\overrightarrow{q_1}) \leq \mathsf{sum}(\overrightarrow{q}) + p_2$, $\mathcal{F}'_1 \subseteq^{\overrightarrow{q_1}}_{grep} \mathcal{F}_1$. Similarly, consider a p_2-family $\mathcal{F}'_2 \subseteq \mathcal{F}_2$ such that for every vector $\overrightarrow{q_2} \geq \overrightarrow{q}$ with $\mathsf{sum}(\overrightarrow{q_2}) \leq \mathsf{sum}(\overrightarrow{q}) + p_1$, $\mathcal{F}'_2 \subseteq^{\overrightarrow{q_2}}_{grep} \mathcal{F}_2$. Then, $\mathcal{F}'_1 \star \mathcal{F}'_2 \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}_1 \star \mathcal{F}_2$.*

Proof. Consider any $B \subseteq V(G)$ of size $\overrightarrow{q}$ for which there is $A \in \mathcal{F}_1 \star \mathcal{F}_2$, such that A fits B. As $A \in \mathcal{F}_1 \star \mathcal{F}_2$, there must exist sets $A_1 \in \mathcal{F}_1$, $A_2 \in \mathcal{F}_2$, such that $A_1 \cup A_2 = A$.

Let $\overrightarrow{\delta_1} = (\delta^1_z = |(A_2 \cap X_z) \setminus B|)_{z \in [k]}$. Note that $|B \cup A_2| = \overrightarrow{q} + \overrightarrow{\delta_1}$, A_1 fits $B \cup A_2$ and $\mathsf{sum}(\overrightarrow{q}) + \mathsf{sum}(\overrightarrow{\delta_1}) \leq \mathsf{sum}(\overrightarrow{q}) + p_2$. By the premise of the lemma, there exist $A'_1 \in \mathcal{F}'_1$ such that A'_1 fits $B \cup A_2$, as $\mathcal{F}'_1 \subseteq^{\overrightarrow{q} + \overrightarrow{\delta_1}}_{grep} \mathcal{F}_1$. The above implies that A_2 fits $B \cup A'_1$, where $A'_1 \in \mathcal{F}'_1$. Let $\overrightarrow{\delta_2} = (\delta^2_z = |(A'_1 \cap X_z) \setminus B|)_{z \in [k]}$, and note that $|B \cup A'_1| = \overrightarrow{q} + \overrightarrow{\delta_2}$, A_2 fits $B \cup A'_1$ and $\mathsf{sum}(\overrightarrow{q}) + \mathsf{sum}(\overrightarrow{\delta_2}) \leq \mathsf{sum}(\overrightarrow{q}) + p_1$. Again, as $\mathcal{F}'_2 \subseteq^{\overrightarrow{q} + \overrightarrow{\delta_2}}_{grep} \mathcal{F}_2$, there exists $A'_2 \in \mathcal{F}'_2$ such that A'_2 fits $B \cup A'_1$. The above discussions imply that, $A'_1 \in \mathcal{F}'_1$, $A'_2 \in \mathcal{F}'_2$, and thus $A'_1 \cup A'_2 \in \mathcal{F}'_1 \star \mathcal{F}'_2$, where $A'_1 \cup A'_2$ fits B. This concludes the proof. ◀

Recall that G is a $K_{i,j}$-free graph, where $i \geq j$. Consider any computable function $f(k)$. Let $\eta_{f(k)} := i \cdot f(k) \cdot k$; where we skip the subscript $f(k)$ when the context is clear. Also, for $p \in \mathbb{N}$, let $\alpha_p := 3 \cdot k \cdot (p\eta)^{i+1}$; again we skip the subscript p, when the context is clear. We next state the main lemma, which lies at the crux of our algorithm.

▶ **Lemma 27** (♠). *Consider any computable function $f : \mathbb{N} \to \mathbb{N} \setminus \{0\}$. There is an algorithm that takes as input $k \in \mathbb{N} \setminus \{0\}$, $p \in \mathbb{N}$, a vector $\overrightarrow{q} = (q_1, q_2, \cdots, q_k)$, and a p-family $\mathcal{F}$ of vertex subsets of a $K_{i,j}$-free graph G on n vertices with a coloring $\chi : V(G) \to [k]$, where $p + \mathsf{sum}(\overrightarrow{q}) \leq f(k)$. In time bounded by $\mathcal{O}(\alpha^{2p + \mathsf{sum}(\overrightarrow{q})} \cdot p \cdot |\mathcal{F}|)$ we can find $\mathcal{F}' \subseteq \mathcal{F}$ with at most $\alpha^{2p + \mathsf{sum}(\overrightarrow{q})}$ sets such that $\mathcal{F}' \subseteq^{\overrightarrow{q}}_{\mathsf{grep}} \mathcal{F}$.*

In the remainder of this section, we prove Theorem 3, assuming the correctness of Lemma 27.

Some Useful Notations. For $z \in \mathbb{N} \setminus \{0\}$ and $z' \in [z]$, let $\gamma_{z,z'}$ be the number of vertices with label z' in the z-Grundy tree (T_z, ℓ_z), i.e., $\gamma_{z,z'} = |\ell_z^{-1}(z')|$.

Let $\overrightarrow{q}^* = \overrightarrow{\gamma_k} := (\gamma_{k,1}, \gamma_{k,2}, \cdots, \gamma_{k,k})$. We will define a vector $\overrightarrow{q_z}^* = (q^*_{z,1}, q^*_{z,2}, \cdots, q^*_{z,k})$, for every $z \in [k]$. Intuitively speaking, the z'th entry of $\overrightarrow{q_z}^*$ will denote the number of vertices with label z' appearing in T_k after removing exactly one subtree rooted at a vertex with label

z. Formally, for each $z' \in \{z+1, z+2, \cdots, k\}$, we have $q^*_{z,z'} = \gamma_{k,z'}$, and for each $z' \in [z]$, $q^*_{z,z'} = \gamma_{k,z'} - \gamma_{z,z'}$. For $z \in [k]$, we let $\overrightarrow{0_z}$ be the vector of dimension k where the zth entry is 1, and all the other entries are 0.

For a tree $\widehat{T}$ rooted at r and $t \in V(\widehat{T})$, we let $\widehat{T}^t$ be the subtree of $\widehat{T}$ rooted at t, i.e., $V(\widehat{T}^t) = \{t' \in V(\widehat{T}) \mid t' = t, \text{ or } t' \text{ is a descendant of } t \text{ in } \widehat{T}\}$ and $\widehat{T}^t = \widehat{T}[V(\widehat{T}^t)]$.

For a set $W \subseteq V(G)$, we say that W is a *k-Grundy set* if there is a k-Grundy witness $\omega : V(T_k) \to W$ for G. Moreover, W is *minimal* if no proper subset $W' \subset W$ is a k-Grundy set for G. For a k-Grundy set W and a k-Grundy witness $\omega : V(T_k) \to W$ for G, for $t \in V(T_k)$, we let $W_{\mathsf{sub},t} = \{\omega(t') \mid t' \in V(T_k^t)\}$ and $W_{\mathsf{exc},t} = \{\omega(t') \mid t' \in V(T_k) \setminus V(T_k^t)\}$.

Recall that we have a graph G and a coloring $\chi : V(G) \to [k]$, where for $z \in [k]$, we have $X_z = \chi^{-1}(z)$. For $z \in [k]$ and $v \in V(G)$, we define $\mathcal{F}_{z,v} := \{W \subseteq \cup_{z' \in [z]} X_{z'} \mid v \in W, W \text{ is } \chi\text{-independent and } z \leq |W| \leq 2^{z-1}\}$.

Description of the Algorithm. The objective of our algorithm will be to compute, for each $z \in [k]$ and $v \in X_z$, a family $\mathcal{F}'_{z,v} \subseteq \mathcal{F}_{z,v}$; starting from $z = 1$ (and then iteratively, for other values of z in increasing order), satisfying the following constraints:

Size Constraint. $|\mathcal{F}'_{z,v}| \leq \alpha^{2^k+1}$.

Correctness Constraint. For any $z \in [k]$ and $v \in X_z$, the following holds:

1. Each $A \in \mathcal{F}'_{z,v}$ is a z-Grundy set in G.
2. Consider any minimal k-Grundy set W, such that $v \in W$ (if it exists). Furthermore, let $\omega : V(T_k) \to W$ be a k-Grundy witness for G. For any $t \in V(T_k)$ with $\omega(t) = v$, where $\overrightarrow{q_t} = |W_{\mathsf{exc},t}|$, there is $W' \in \mathcal{F}'_{z,v} \subseteq \mathcal{F}_{z,v}$ such that $W_{\mathsf{exc},t} \cup W'$ is a k-Grundy set in G, i.e., $\mathcal{F}'_{z,v} \subseteq^{\overrightarrow{q_t}}_{grep} \mathcal{F}_{z,v}$.

Base Case. We are in our base case when $z = 1$; note that $[2^0] = \{1\}$. For each $v \in X_1$, set $\mathcal{F}'_{1,v} := \mathcal{F}_{1,v} = \{\{v\}\}$. Note that $\mathcal{F}'_{1,v}$ satisfies both the size and the correctness constraints.

Recursive Formula. Consider $z \in [k] \setminus \{1\}$ and $v \in X_z$. We suppose that for each $z' \in [z-1]$ and $v' \in X_{z'}$, we have computed $\mathcal{F}'_{z',v'}$ that satisfies both the size and the correctness constraints.

For each $z' \in [z-1]$, we create a family $\mathcal{F}_{z,v,z'}$, initialized to $\emptyset$ as follows. For each $u \in X_{z'} \cap N_G(v)$ and $W \in \mathcal{F}'_{z',u}$, if $W \cup \{v\}$ is χ-independent and $|W \cup \{v\}| \leq 2^{z-1}$, then add $W \cup \{v\}$ to $\mathcal{F}_{z,v,z'}$. Note that $|\mathcal{F}_{z,v,z'}| \leq n \cdot \alpha_p^{2^k+1}$, where $p = 2^{z'-1}$. Using Lemma 27, for each vector $\overrightarrow{q} \leq \overrightarrow{q_z}^*$, we compute $\mathcal{F}'_{z,v,z',\overrightarrow{q}} \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}_{z,v,z'}$, where $|\mathcal{F}'_{z,v,z',\overrightarrow{q}}| \leq \alpha^{2^k}$, and set $\mathcal{F}'_{z,v,z'} = \cup_{\overrightarrow{q} \leq \overrightarrow{q_z}} \mathcal{F}'_{z,v,z',\overrightarrow{q}}$. Note that $\mathcal{F}'_{z,v,z'} \leq 2^{(k-1)k} \cdot \alpha^{2^k} \leq \alpha^{2^k+1}$ and we can compute it in time bounded by $\mathcal{O}(\alpha^{2^k+1} \cdot 2^{k-1} \cdot |\mathcal{F}_{z,v,z'}|)$.

Next we will iteratively "combine and reduce" the families $\mathcal{F}'_{z,v,z'}$, for $z' \in [z]$, to obtain a family $\widehat{\mathcal{F}}_{z,v} \subseteq \mathcal{F}_{z,v}$ as follows. We set $\widehat{\mathcal{F}}_{z,v,1} := \mathcal{F}'_{z,v,1}$. Iteratively, (in increasing order), for each $z' \in [z-1] \setminus \{1\}$, we do the following:

1. Set $\widetilde{\mathcal{F}}_{z,v,z'} := \widehat{\mathcal{F}}_{z,v,z'-1} \star \mathcal{F}'_{z,v,z'}$.
2. Compute $\widehat{\mathcal{F}}_{z,v,z',\overrightarrow{q}} \subseteq^{\overrightarrow{q}}_{grep} \widetilde{\mathcal{F}}_{z,v,z'}$, for each $\overrightarrow{q} \leq \overrightarrow{q_{z'}} = \overrightarrow{\gamma_k} - \left(\sum_{\widehat{z} \in [z']} (\overrightarrow{\gamma_k} - \overrightarrow{q_z}^*)\right) - \overrightarrow{0_z}$, and set $\widehat{\mathcal{F}}_{z,v,z'} = \cup_{\overrightarrow{q} \leq \overrightarrow{q_{z'}}} \widehat{\mathcal{F}}_{z,v,z',\overrightarrow{q}}$. Note that $|\widehat{\mathcal{F}}_{z,v,z'}| \leq \alpha^{2^k+1}$ and it can be computed in time bounded by $\mathcal{O}(\alpha^{2^k+1} \cdot 2^{k-1} \cdot |\widetilde{\mathcal{F}}_{z,v,z'}|)$.

We add each $A \in \widehat{\mathcal{F}}_{z,v,z-1}$ to $\mathcal{F}'_{z,v}$, which is a z-Grundy set in G (note that since the size of each set is bounded by 2^{k-1}, we can easily do it in the allowed amount of time). In the following lemma, we show that $\mathcal{F}'_{z,v}$ satisfies the correctness constraints.

▶ **Lemma 28.** $\mathcal{F}'_{z,v}$ *satisfies the correctness constraint.*

Proof. Consider any $v \in X_z$ and a minimal k-Grundy set W, such that $v \in W$ (if it exists) and let $\omega : V(T_k) \to W$ be a k-Grundy witness for G. Next consider any $t \in V(T_k)$ with $\omega(t) = v$. We will argue that, there is $W' \in \mathcal{F}'_{z,v}$ such that $W_{\mathsf{exc},t} \cup W'$ is a k-Grundy set in G. For each $z' \in [z-1]$, let $t_{z'}$ be the child of t in T_k with $\ell_k(t_{z'}) = z'$ and $v_{z'} = \omega(t_{z'})$. Note that for each $z' \in [z-1]$, $v_{z'} \in X_{z'}$. For each $z' \in [z-1]$, let $\widehat{A}_{z'} = \{\omega(t') \mid t' \in V(T_k^{t_{z'}})\}$, and note that $|\widehat{A}_{z'}| \leq 2^{z'-1}$. Now we iteratively take the union of the above sets as follows. For each $z' \in [z-1]$, let $A_{z'} = \cup_{\widehat{z} \in [z']} \widehat{A}_{\widehat{z}}$. Now for each $z' \in [z-1]$, we construct a subset, $B_{z'}$ of W that contains $\omega(t')$, for each $t' \in V(T_k)$ that does not belong to the subtrees rooted at any of the vertices $t_1, t_2, \cdots, t_{z'}$. Formally, for $z' \in [z-1]$, let $B_{z'} = \{\omega(t') \mid t' \in V(T_k) \setminus \big(\bigcup_{z'' \in [z']} V(T_k^{t_{z''}})\big)\}$. Furthermore, let $\overrightarrow{s_{z'}}$ be the size of $B_{z'}$. Notice that for each $z' \in [z-1]$, all of the following holds:

1. $\overrightarrow{s_{z'}} \leq \overrightarrow{q_{z'}}$,
2. $|A_{z'}| \leq \sum_{\widehat{z} \in [z']} 2^{\widehat{z}-1}$,
3. $|\widehat{A}_{z'}| \leq 2^{z'-1}$ and $\widehat{A}_{z'} \in \mathcal{F}_{z',v_{z'}}$,
4. $A_{z'} \cup B_{z'} = W$, and thus, $A_{z'}$ fits $B_{z'}$.

We will now iteratively define sets $A'_1, A'_2, \cdots, A'_{z-1}$ and functions $\omega_1, \omega_2 \cdots, \omega_{z-1}$, and we will ensure that, for each $z' \in [z-1]$, we have: i) $\omega_{z'} : V(T_k) \to A'_{z'} \cup B_{z'}$ is a k-Grundy witness for G, ii) for each $z' \in [z-1]$ and $t' \in V(T_k) \setminus \big(\bigcup_{z'' \in [z']} V(T_k^{t_{z''}})\big)$, we have $\omega_{z'}(t') = \omega(t')$, iii) $A'_{z'} \in \widehat{\mathcal{F}}_{z,v,z'}$, and iv) for each $z'' \in [z']$, there is a minimal z''-Grundy set $A'_{z',z''} \subseteq A'_{z'}$, where the unique vertex in $A'_{z',z''} \cap X_{z''}$ is a neighbor of v.

Recall that $z \geq 2$ and $\widehat{\mathcal{F}}_{z,v,1} = \mathcal{F}'_{z,v,1} \subseteq \mathcal{F}_{z,v,1}$. Also, we have $\widehat{A}_1 = A_1 = \{u'\}$, for some $u' \in N_G(v) \cap X_1$, and $A_1 \cup B_1 = W$ is a k-Grundy set. Thus, there must exist $A'_1 \in \mathcal{F}'_{z,v,1}$ such that $A'_1 \cup B_1$ is χ-independent. Moreover by the construction of $\mathcal{F}'_{z,v,1}$, $A'_1 = \{u\}$, for some $u \in N_G(v) \cap X_1$. Let $\omega_1 : V(T_k) \to A'_1 \cup B_1$ be the function such that for each $t' \in V(T_k) \setminus \{t_1\}$, we have $\omega_1(t') = \omega(t')$ and $\omega_1(t_1) = u$. As $A'_1 \cup B_1$ is χ-independent and $\{u, v\} \in E(G)$, we can obtain that ω_1 is a k-Grundy witness for G. Note that if $z = 2$, then by the above arguments, we have constructed the desired sets and functions, which is just the set A'_1 and the function ω_1.

We now consider the case when $z' \geq 2$. Also, we assume that for some $\widehat{z} \in [z-2]$, for each $z' \in [\widehat{z}]$, we have constructed $A'_{z'}$ and $\omega_{z'}$ satisfying the desired condition. Now we prove the statement for $z' = \widehat{z} + 1$. Note that $A'_{z'-1} \cup B_{z'-1}$ is a k-Grundy set and $\omega_{z'-1} : V(T_k) \to A'_{z'-1} \cup B_{z'-1}$ is a k-Grundy witness for G, where $A'_{z'-1} \in \widehat{\mathcal{F}}_{z,v,z'}$. Note that $\widehat{A}_{z'} \subseteq B_{z'-1}$. Let $B'_{z'} = \{\omega_{z'-1}(t') \mid t' \in V(T_k) \setminus V(T_k^{t_{z'}})\}$. Note that $|B'_{z'}| \leq \overrightarrow{q_{z'}}$ and $\widehat{A}_{z'}$ fits $B'_{z'}$, and recall that $\widehat{A}_{z'} \in \mathcal{F}_{z',v_{z'}}$. As $\mathcal{F}'_{z',v_{z'}} \subseteq^{\overrightarrow{q}}_{grep} \mathcal{F}_{z',v_{z'}}$, for every $\overrightarrow{q} \leq \overrightarrow{q_{z'}}$, there must exists $\widetilde{A}_{z'} \in \mathcal{F}'_{z',v_{z'}}$, such that $\widetilde{A}_{z'}$ fits $B'_{z'}$. By the construction of $\mathcal{F}'_{z',v_{z'}}$, we have $v_{z'} \in \widetilde{A}_{z'}$ and $\widetilde{A}_{z'} \cup B'_{z'}$ is χ-independent, and also $v \in B'_{z'}$. From the above discussions we can conclude that $\widetilde{A}_{z'} \cup \{v\} \in \mathcal{F}_{z,v,z'}$. Moreover, as $A'_{z'-1} \in \widehat{\mathcal{F}}_{z,v,z'-1}$, $A'_{z'-1} \subseteq B'_{z'}$ and $\widetilde{A}_{z'} \cup B'_{z'}$ is χ-independent, we can obtain that $A'_{z'-1} \cup \widetilde{A}_{z'} \cup \{v\} \in \widehat{\mathcal{F}}_{z,v,z'-1} \star \mathcal{F}'_{z,v,z'} = \widetilde{\mathcal{F}}_{z,v,z'}$. As $\widehat{\mathcal{F}}_{z,v,z'} \subseteq^{\overrightarrow{q}}_{grep} \widetilde{\mathcal{F}}_{z,v,z'}$, for every $\overrightarrow{q} \leq \overrightarrow{q_{z'}}^*$ and $|B_{z'}| \leq \overrightarrow{q_{z'}}^*$, there must exists $A'_{z'} \in \widehat{\mathcal{F}}_{z,v,z'}$ such that $A'_{z'} \cup B_{z'}$ is χ-independent.

As $A'_{z'} \in \widehat{\mathcal{F}}_{z,v,z'}$, there must exist $\widehat{C} \in \widehat{\mathcal{F}}_{z,v,z'-1}$ and $C' \cup \{v\} \in \mathcal{F}'_{z,v,z'}$, such that $A'_{z'} = \widehat{C} \cup C' \cup \{v\}$. By the correctness for $z'-1$, C' contains for each $z'' \in [z'-1]$, a minimal z''-Grundy set $A'_{z'-1,z''} \subseteq A'_{z'-1}$, where the unique vertex in $A'_{z'-1,z''} \cap X_{z''}$ is a neighbor of v. Also by the construction of $\mathcal{F}'_{z,v,z'}$, C' contains a minimal z'-Grundy set C'' in G, where the unique vertex in $C'' \cap X_{z'}$ is a neighbor of v. From the above discussions, we can conclude that $A'_{z'} \cup B_{z'}$ is a k-Grundy set in G. ◀

Using the above algorithm, we can compute for each $z \in [k]$ and $v \in X_z$, a family $\mathcal{F}'_{z,v} \subseteq \mathcal{F}_{z,v}$ that satisfies the correctness and the size constraints, in time bounded by $\alpha^{\mathcal{O}(2^k+1)} \cdot n^{\mathcal{O}(1)}$. Note that G has a Grundy coloring using at least k colors if and only if for some $v \in X_k$, $\mathcal{F}'_{z,v} \neq \emptyset$. This implies a proof of Theorem 3.

References

1 Pierre Aboulker, Édouard Bonnet, Eun Jung Kim, and Florian Sikora. Grundy coloring and friends, half-graphs, bicliques. *Algorithmica*, pages 1–28, 2022. doi:10.1007/S00453-022-01001-2.

2 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995. doi:10.1145/210332.210337.

3 Júlio César Silva Araújo and Cláudia Linhares Sales. Grundy number on p_4-classes. *Electron. Notes Discret. Math.*, 35:21–27, 2009. doi:10.1016/J.ENDM.2009.11.005.

4 R. Balakrishnan and T Kavaskar. Interpolation theorem for partial grundy coloring. *Discrete Mathematics*, 313(8):949–950, 2013. doi:10.1016/J.DISC.2013.01.018.

5 Rangaswami Balakrishnan and T. Kavaskar. Color chain of a graph. *Graphs Comb.*, 27(4):487–493, 2011. doi:10.1007/S00373-010-0989-7.

6 Rémy Belmonte, Eun Jung Kim, Michael Lampis, Valia Mitsou, and Yota Otachi. Grundy distinguishes treewidth from pathwidth. In *28th Annual European Symposium on Algorithms (ESA 2020)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020.

7 Édouard Bonnet, Florent Foucaud, Eun Jung Kim, and Florian Sikora. Complexity of grundy coloring and its variants. *Discret. Appl. Math.*, 243:99–114, 2018. doi:10.1016/J.DAM.2017.12.022.

8 Gregory J. Chaitin, Marc A. Auslander, Ashok K. Chandra, John Cocke, Martin E. Hopkins, and Peter W. Markstein. Register allocation via coloring. *Comput. Lang.*, 6(1):47–57, 1981. doi:10.1016/0096-0551(81)90048-5.

9 CWK Chen and David YY Yun. Unifying graph-matching problem with a practical solution. In *Proceedings of International Conference on Systems, Signals, Control, Computers*, volume 55, 1998.

10 C. A. Christen and S. M. Selkow. Some perfect coloring properties of graphs. *Journal of Combinatorial Theory, Series B*, 27(1):49–59, 1979. doi:10.1016/0095-8956(79)90067-4.

11 Marek Cygan, Fedor V Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized algorithms*, volume 5. Springer, 2015. doi:10.1007/978-3-319-21275-3.

12 Lyes Dekar, Brice Effantin, and Hamamache Kheddouci. An incremental distributed algorithm for a partial grundy coloring of graphs. In Ivan Stojmenovic, Ruppa K. Thulasiram, Laurence Tianruo Yang, Weijia Jia, Minyi Guo, and Rodrigo Fernandes de Mello, editors, *Parallel and Distributed Processing and Applications, 5th International Symposium, ISPA 2007, Niagara Falls, Canada, August 29-31, 2007, Proceedings*, volume 4742 of *Lecture Notes in Computer Science*, pages 170–181. Springer, 2007. doi:10.1007/978-3-540-74742-0_18.

13 Reinhard Diestel. *Graph theory*, volume 173. Springer-Verlag Heidelberg, 2017.

14 Irit Dinur, Elchanan Mossel, and Oded Regev. Conditional hardness for approximate coloring. *SIAM J. Comput.*, 39(3):843–873, 2009. doi:10.1137/07068062X.

15 B. Effantin, N. Gastineau, and O. Togni. A characterization of b-chromatic and partial grundy numbers by induced subgraphs. *Discrete Mathematics*, 339(8):2157–2167, 2016. doi:10.1016/J.DISC.2016.03.011.

16 P. Erdös, S. T. Hedetniemi, R. C. Laskar, and G. C. E. Prins. On the equality of the partial grundy and upper ochromatic numbers of graphs. *Discrete Mathematics*, 272(1):53–64, 2003. doi:10.1016/S0012-365X(03)00184-5.

17 P Fahad. *Dynamic Programming using Representative Families.* PhD thesis, Homi Bhabha National Institute, 2015.

18 Uriel Feige and Joe Kilian. Zero knowledge and the chromatic number. *J. Comput. Syst. Sci.*, 57(2):187–199, 1998. doi:10.1006/JCSS.1998.1587.

19 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, 2016. doi:10.1145/2886094.

20 N. Goyal and S. Vishwanathan. Np-completeness of undirected grundy numbering and related problems. *Unpublished manuscript*, 1997.

21 P. M. Grundy. Mathematics and games. *Eureka*, 2:6–9, 1939.

22 András Gyárfás, Zoltán Király, and Jenö Lehel. On-line 3-chromatic graphs i. triangle-free graphs. *SIAM J. Discret. Math.*, 12(3):385–411, 1999. doi:10.1137/S089548019631030X.

23 F. Havet and L. Sampaio. On the grundy and b-chromatic numbers of a graph. *Algorithmica*, 65(4):885–899, 2013. doi:10.1007/S00453-011-9604-4.

24 S. M. Hedetniemi, S. T. Hedetniemi, and T. Beyer. A linear algorithm for the grundy (coloring) number of a tree. *Congressus Numerantium*, 36:351–363, 1982.

25 Allen Ibiapina and Ana Silva. b-continuity and partial grundy coloring of graphs with large girth. *Discrete Mathematics*, 343(8):111920, 2020. doi:10.1016/J.DISC.2020.111920.

26 Hal A. Kierstead and Karin Rebecca Saoub. First-fit coloring of bounded tolerance graphs. *Discret. Appl. Math.*, 159(7):605–611, 2011. doi:10.1016/J.DAM.2010.05.002.

27 Guy Kortsarz. A lower bound for approximating the grundy number. *Discrete Mathematics & Theoretical Computer Science*, 9, 2007.

28 Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, Roohani Sharma, and Meirav Zehavi. Covering small independent sets and separators with applications to parameterized algorithms. *ACM Transactions on Algorithms (TALG)*, 16(3):1–31, 2020. doi:10.1145/3379698.

29 Dániel Marx. Graph colouring problems and their applications in scheduling. *Periodica Polytechnica Electrical Engineering (Archives)*, 48(1-2):11–16, 2004.

30 Dániel Marx. A parameterized view on matroid optimization problems. *Theor. Comput. Sci.*, 410(44):4471–4479, 2009. doi:10.1016/J.TCS.2009.07.027.

31 Moni Naor, Leonard J Schulman, and Aravind Srinivasan. Splitters and near-optimal derandomization. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 182–191. IEEE, 1995. doi:10.1109/SFCS.1995.492475.

32 B. S. Panda and Shaily Verma. On partial grundy coloring of bipartite graphs and chordal graphs. *Discret. Appl. Math.*, 271:171–183, 2019. doi:10.1016/J.DAM.2019.08.005.

33 L. Sampaio. *Algorithmic aspects of graph colourings heuristics*. PhD thesis, University Nice Sophia Antipolis, 2012.

34 Z. Shi, W. Goddard, S. T. Hedetniemi, K. Kennedy, R. Laskar, and A. McRae. An algorithm for partial grundy number on trees. *Discrete Mathematics*, 304(1-3):108–116, 2005. doi:10.1016/J.DISC.2005.09.008.

35 Zixing Tang, Baoyindureng Wu, Lin Hu, and Manouchehr Zaker. More bounds for the grundy number of graphs. *J. Comb. Optim.*, 33(2):580–589, 2017. doi:10.1007/S10878-015-9981-8.

36 Shaily Verma and B. S. Panda. Grundy coloring in some subclasses of bipartite graphs and their complements. *Information Processing Letters*, 163:105999, 2020. doi:10.1016/J.IPL.2020.105999.

37 M. Zaker. Grundy chromatic number of the complement of bipartite graphs. *Australasian Journal of Combinatorics*, 31:325–330, 2005. URL: http://ajc.maths.uq.edu.au/pdf/31/ajc_v31_p325.pdf.

38 M. Zaker. Results on the grundy chromatic number of graphs. *Discrete mathematics*, 306(23):3166–3173, 2006. doi:10.1016/J.DISC.2005.06.044.

39 Manouchehr Zaker. Inequalities for the grundy chromatic number of graphs. *Discret. Appl. Math.*, 155(18):2567–2572, 2007. doi:10.1016/J.DAM.2007.07.002.