Harmonious Coloring: Parameterized Algorithms and Upper Bounds

Sudeshna Kolay[1], Ragukumar Pandurangan[1], Fahad Panolan[2], Venkatesh Raman[1], and Prafullkumar Tale[1](✉)

[1] The Institute of Mathematical Sciences, Chennai, India
{skolay,ragukumar,vraman,pptale}@imsc.res.in
[2] University of Bergen, Bergen, Norway
Fahad.Panolan@ii.uib.no

Abstract. A harmonious coloring of a graph is a partitioning of its vertex set into parts such that, there are no edges inside each part, and there is at most one edge between any pair of parts. It is known that finding a minimum harmonious coloring number is NP-hard even in special classes of graphs like trees and split graphs.

We initiate a study of parameterized and exact exponential time complexity of harmonious coloring. We consider various parameterizations like by solution size, by above or below known guaranteed bounds and by the vertex cover number of the graph. While the problem has a simple quadratic kernel when parameterized by the solution size, our main result is that the problem is fixed-parameter tractable when parameterized by the size of a vertex cover of the graph. This is shown by reducing the problem to multiple instances of fixed variable integer linear programming.

We also observe that it is $W[1]$-hard to determine whether at most $n - k$ or $\Delta + 1 + k$ colors are sufficient in a harmonious coloring of an n-vertex graph G, where Δ is the maximum degree of G and k is the parameter. Concerning exact exponential time algorithms, we develop a $2^n n^{\mathcal{O}(1)}$ algorithm for finding a minimum harmonious coloring in split graphs improving on the naive $2^{\mathcal{O}(n \log n)}$ algorithm.

1 Introduction and Motivation

Graph Coloring is the problem of partitioning the vertex set of a graph to satisfy some constraints. Coloring problems have been extensively studied in discrete mathematics and theoretical computer science. Given a coloring χ of a graph G, the set of vertices that receive the same color is said to be a *color class*. One of the most well-known coloring problems is the chromatic number problem that seeks the minimum number of colors required so that each color class induces an independent set (i.e. no pair of vertices in a set is adjacent), and it is one of Karp's 21 NP-complete problems from 1972 [18]. Lawler gave an algorithm for the problem running in time $2.4423^n n^{\mathcal{O}(1)}$ on an n-vertex graph [19]. Later, using the principle of inclusion-exclusion Björklund et al. [5] gave an algorithm

P. Heggernes (Ed.): WG 2016, LNCS 9941, pp. 245–256, 2016.
DOI: 10.1007/978-3-662-53536-3_21

running in time $2^n n^{\mathcal{O}(1)}$ on an n-vertex graph and this is the fastest known exact algorithm for the problem.

Different variants of the graph coloring problem have been studied in the literature. The Achromatic Number seeks the *maximum* number of colors required so that each color class induces an independent set, and there is at least one edge between every pair of color classes. A characterization for this problem was given in [14] using which one can obtain an FPT algorithm for Achromatic Number parameterized by the solution size (see Sect. 2.1 for definitions on parameterized complexity). The Pseudo-Achromatic Number problem is a generalization of Achromatic Number, and does not demand that each color class induces an independent set. This problem is also FPT parameterized by the solution size [7]. Another related problem is the b-Chromatic Number. Here the objective is to color the vertices with the same properties as that in Achromatic Number, but insist that in each color class there is a vertex that has a neighbor in every other color class. This problem was introduced in [2]. The problem is W[1]-hard when parameterized by the solution size [22].

In 1989, Hopcroft and Krishnamoorthy [15] introduced the notion of harmonious coloring. A harmonious coloring of a graph is a partition of the vertex set into sets such that every set induces an independent set and additionally between any pair of sets, there is *at most* one edge. The minimum number of sets in such a partition is called the harmonious coloring number of the graph. Determining whether a graph has harmonious coloring using at most k colors is known to be NP-complete [15], even in trees [13], split graphs [3], interval graphs [3,6] and several other classes of graphs [3,4,6,12,13,16]. Polynomial time algorithms are known for some special classes of graphs [21], the most important being for trees of bounded degree [11].

In this paper, we initiate the parameterized complexity of the problem under natural parameterizations. With solution size k (the harmonious coloring number) as a parameter, there is a trivial kernel on $O(k^2)$ vertices and edges, and this is discussed in Sect. 4.1. In this section, we also discuss parameterized complexity of parameterizing above or below some known bounds for harmonious coloring number. As the problem is NP-complete on trees, the problem parameterized by the treewidth or feedback vertex set is trivially para NP-hard. Our main result is that the problem is fixed-parameter tractable when parameterized by the size of the minimum vertex cover of the graph. This is shown by solving several bounded variable integer linear programming (ILP) problems. The number of ILPs is upper bounded by a function of minimum vertex cover. This is developed in Sect. 4.2. In Sect. 5, we discuss exact exponential algorithms for harmonious coloring, and give an $2^{\mathcal{O}(n)}$ algorithm in split graphs, improving on the naive $2^{\mathcal{O}(n \log n)}$ algorithm. In Sect. 3, we develop improved upper bounds on the harmonious coloring number in terms of the vertex cover number and the maximum degree of the graph. Results marked with a ($\star$) have their proofs in the full version of this paper.

2 Preliminaries

We use $\mathbb{N}$ and $\mathbb{Z}$ to denote the set of natural numbers and set of integers, respectively. For $n \in \mathbb{N}$ we use $[n]$ to denote $\{1, \ldots, n\}$. We use standard notations from graph theory [9]. By "graph" we mean simple undirected graph. The vertex set and edge set of a graph G are denoted as $V(G)$ and $E(G)$ respectively. The complement of a graph G, denoted by $\overline{G}$, has $V(G)$ as its vertex set and $\binom{V(G)}{2} \setminus E(G)$ as its edge set. Here, $\binom{V(G)}{2}$ denotes the family of two sized subsets of $V(G)$. The *neighborhood* of a vertex v is represented as $N_G(v)$, or, when the context of the graph is clear, simply as $N(v)$. The *closed neighborhood* of a vertex v, denoted by $N[v]$, is the subset $N(v) \cup \{v\}$. For set U, we define $N(U)$ as union of $N(v)$ all vertices v in U. If $U = \emptyset$ then $N(U) = \emptyset$. For two disjoint subsets $V_1, V_2 \subseteq V(G)$, $E(V_1, V_2)$ is set of edges where one end point is in V_1 and another is in V_2. An edge in the set $E(V_1, V_2)$ is said to be *going across*. A *trivial* component of graph is a component which does not contain any edge. A *non-trivial component* of a graph is a connected component of G that has at least two vertices. The function $d_G : V(G) \times V(G) \to \mathbb{N}$ corresponds to the minimum distance between a pair of vertices in the graph G. A d-degenerate graph is a graph G where $V(G)$ has an ordering in which any vertex has at most d neighbors with indices lower than that of the vertex. For a graph G, a set $S \subseteq V(G)$ is called a *vertex cover* of G if $G - S$ is an independent set. A graph G is called a *split graph* if $V(G)$ has a bipartition (V_1, V_2) such that $G[V_1]$ is an induced clique and $G[V_2]$ is an induced independent set. In this case, $(G[V_1], G[V_2])$ is called a *split partition* of G. No split graph contains a 4-cycle (C_4), a 5-cycle (C_5) or the complement of a 4-cycle ($2K_2$) as an induced subgraph. The finite set of graphs $\{C_4, C_5, 2K_2\}$ is said to be a *finite forbidden set* of the class of split graphs. Each graph in the finite forbidden set is referred to as a *forbidden structure*.

A function $h : V(G) \to [k]$, where k is a positive integer, is called a *coloring function*. For a coloring function h and for any $i \in [k]$, the vertex subset $h^{-1}(i)$ is called the i^{th} *color class* of h. If no edge has both its end points in the same color class then coloring function is said to be *proper*. *Harmonious coloring* is a proper coloring with additional property that there is at most one edge across any two color classes. The minimum number of colors required for a harmonious coloring of a graph G is denoted by $\mathsf{hc}(G)$. The restriction of a coloring function h to a subset $V' \subseteq V(G)$, denoted by $h|_{V'}$, is a coloring function such that $h|_{V'} : V' \to [k]$, and $h|_{V'}(u) = h(u)$ for each vertex $u \in V'$. In this case, h is said to be an *extension* of $h|_{V'}$. For a subset $V' \subseteq V(G)$, $h(V') = \{i | h^{-1}(i) \cap V' \neq \emptyset\}$.

The technical tool we use to prove that Harmonious Coloring is fixed-parameter tractable (defined in next section) by size of vertex cover is the fact that Integer Linear Programming is fixed-parameter tractable parameterized by the number of variables. An instance of Integer Linear Programming consists of a matrix $A \in \mathbb{Z}^{m \times p}$, a vector $b \in \mathbb{Z}^m$ and a vector $c \in \mathbb{Z}^p$. The goal is to find a vector $x \in \mathbb{Z}^p$ which satisfies $Ax \leq b$ and minimizes the value of $c \cdot x$ (scalar product of c and x). We assume that an input is given in binary and thus the size of the input is the number of bits in its binary representation.

Proposition 1 **([17], [20]).** *An* Integer Linear Programming *instance of size L with p variables can be solved using $\mathcal{O}(p^{2.5p+o(p)} \cdot (L + \log M_x) \cdot \log(M_x \cdot M_c))$ arithmetic operations and space polynomial in $L + \log M_x$, where M_x is an upper bound on the absolute value a variable can take in a solution, and M_c is the largest absolute value of a coefficient in the vector c.*

2.1 Parameterized Complexity

The goal of parameterized complexity is to find ways of solving NP-hard problems more efficiently than brute force by associating a *small* parameter to each instance. Formally, a *parameterization* of a problem is assigning a positive integer parameter k to each input instance and we say that a parameterized problem is *fixed-parameter tractable (FPT)* if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{\mathcal{O}(1)}$, where $|I|$ is the size of the input and f is an arbitrary computable function depending only on the parameter k. Such an algorithm is called an FPT algorithm and such a running time is called FPT running time. There is also an accompanying theory of hardness using which one can identify parameterized problems that are unlikely to admit FPT algorithms. The hard classes are $W[i], i \in \mathbb{N}$. For the purpose of this paper, it is enough to know that the Independent Set problem is W[1]-hard [10].

A parameterized problem is said to be in the class para-NP if it has a nondeterministic algorithm with FPT running time. To show that a problem is para-NP-hard, we need to show that the problem is NP-hard when the parameter takes a value from a finite set of positive integers.

Another direction of research is in providing a refinement of the FPT class, through the concept of *kernelization*. A parameterized problem is said to admit a $h(k)$-*kernel* if there is a polynomial time algorithm (the degree of the polynomial is independent of k), called a *kernelization* algorithm, that reduces the input instance to an instance with size upper bounded by $h(k)$, while preserving the answer. If the function $h(k)$ is polynomial in k, then we say that the problem admits a polynomial kernel. For more on parameterized complexity, see the recent book [8].

3 Upper and Lower Bounds and Structural Results

In this section, we give some general upper bounds of harmonious coloring number based on other natural graph parameters and show some structural results which are used later in our algorithms.

Observation 1. *For a given graph G and two vertices u, v, if u and v belong to the same harmonious color class then $d_G(u, v) > 2$.*

Definition 1 (Identify). *For a graph G, identifying a vertex set U of $V(G)$ is the operation of deleting U, adding a new vertex w and the edge set $\{wx | x \notin U, \exists u \in U \, and \, xu \in E(G)\}$.*

Observation 2 (⋆). *For a graph G, let ϕ be an optimal harmonious coloring. Suppose the graph G' is formed by identifying a color class of ϕ. Then* $\mathsf{hc}(G) = \mathsf{hc}(G')$.

Lemma 1 (⋆). *Let G be a graph without isolated vertices, X be a vertex cover of G, and let H be the auxiliary graph defined such that $V(H) = V(G - X)$ and for $u, v \in V(H)$, $uv \in E(H)$ if $d_G(u, v) = 2$. A coloring function h of G, where (1) $h(X) \cap h(V(G-X)) = \emptyset$, (2) $h(i) \neq h(j)$ for all $i \neq j \in X$, is a harmonious coloring of G if and only if $h|_{V(G-X)}$ is a proper coloring of H.*

Let $\Delta(G)$ denote the maximum degree of the graph, and $vc(G)$ denote the vertex cover number of G. We use Δ if the graph G is clear from the context. We show the following bound for general graphs.

Theorem 1. *For any graph G with $\Delta \geq 2, \Delta + 1 \leq \mathsf{hc}(G) \leq vc(G) + \Delta(\Delta - 1)$.*

Proof. By Observation 1, any two vertices in the same harmonious color class should be at a distance three or more from each other. This implies that for any vertex u, every vertex in its closed neighbourhood gets a separate color. Since this is true for a vertex with the highest degree, lower bound on harmonious coloring follows.

We first construct a harmonious coloring with $vc(G) + \Delta(\Delta - 1) + 1$ many colors and then apply a trick to save one color. Let X be a vertex cover of graph G. Construct a coloring $\phi : V(G) \rightarrow [vc(G) + \Delta(\Delta - 1) + 1]$ in the following fashion: Color each vertex in vertex cover X with separate color which will not be used for remaining vertices. Construct an auxiliary graph H as mentioned in Lemma 1. Notice that $\Delta(H) = \Delta(G)(\Delta(G) - 1)$. Graph H can be properly colored using $\Delta(H) + 1$ many colors ([9] p.115). Coloring $\phi|_{V(G-X)}$ is proper coloring of H and satisfies the premises of Lemma 1 hence it is harmonious coloring of G.

We now show how to save one color from this coloring using a similar idea from [1]. Let X be the vertex cover. If our greedy coloring above used only $\Delta(\Delta-1)$ colors to color vertices of $V(G)\setminus X$, then we are already done. Otherwise, pick any vertex u in X. We recolor u using a color used by vertices in $V(G) \setminus X$. Let u be adjacent to $i \leq \Delta - 1$ vertices in X (If all neighbors of u are in X, then u can be moved out of X, without loss of generality). Hence there are at most $i(\Delta - 1)$ vertices in $V(G) \setminus X$ which are at distance two from vertex u. There are at most $\Delta - i$ vertices adjacent to u in $V(G) \setminus X$. Colors used by all these vertices can not be used to recolor vertex u because of Observation 1 but u can be colored with any other color. Thus the number of forbidden colors is $i(\Delta - 1) + \Delta - i = i(\Delta - 2) + \Delta$. But $i(\Delta - 2) + \Delta \leq (\Delta - 1)(\Delta - 2) + \Delta = \Delta(\Delta - 1) - \Delta + 2 \leq \Delta(\Delta - 1)$ when $\Delta \geq 2$ and hence we can always find a color to recolor vertex u reducing the upper bound by 1. □

The upper bound is tight for C_4, a cycle on 4 vertices.

Theorem 2 (⋆). *If G is a d-degenerate graph, then $\Delta + 1 \leq \mathsf{hc}(G) \leq vc(G) + d(\Delta - 1) + \Delta(d - 1) + 1$.*

The following corollary follows from Theorem 2 as a forest is 1-degenerate.

Corollary 1. *If G is a forest with at least one edge, then $\Delta + 1 \leq \mathsf{hc}(G) \leq vc(G) + \Delta$.*

The upper bounds in Theorem 1 and Corollary 1 improve respectively the bounds of Theorems 6 and 4 of [1].

4 Parameterized Complexity of Harmonious Coloring

4.1 'Standard' and 'Above/Below Guarantee' Parameterizations

In this subsection, we capture some easy observations on the parameterized complexity of harmonious coloring under some standard parameterizations. We start with the following theorem whose proof (given in the full version of this paper) follows from the observation that if the number of edges is 'large', then the harmonious coloring number has to be large.

Lemma 2 ($\star$). *Let G be a graph on n vertices and m edges.* Harmonious Coloring*, parameterized by the number of colors used, is* FPT*with a quadratic kernel.*

The proof of the above theorem suggests that the harmonious coloring number of most graphs is large with respect to the number of vertices. The number of vertices n is a trivial upper bound and Theorem 1 gives a lower bound of $\Delta+1$ for the harmonious coloring number of a graph. So the natural question is: is it FPT to determine whether one can harmoniously color using at most $n-k$ or $\Delta + k + 1$ colors where the parameter is k. We prove the following theorem.

Theorem 3 ($\star$). *(i) It is W[1]-hard to determine whether a given n-vertex graph has harmonious coloring number at most $n - k$ where k is the parameter. (ii) It is para-NP-hard to determine whether a given graph has a harmonious coloring number at most $\Delta + 1 + k$ where Δ is the maximum degree of the graph, and k is the parameter.*

4.2 Parameterization by Size of Vertex Cover

As the Harmonious coloring is NP-complete on trees, it is trivially para NP-hard when parameterized by the treewidth of the graph or the feedback vertex set size of the graph. In this section, we consider the structural parameterization by the well-studied vertex cover number of the graph. We describe an FPT algorithm for Harmonious coloring when parameterized by the size of a vertex cover of the input graph. We show that the problem reduces to several instances of Integer Linear Programming. We assume that the input graph G has no isolated vertices. Otherwise, for any harmonious coloring of the input graph G, we can include the set of isolated vertices into any one of the color classes.

In case of structural parameters, sometimes it is necessary to demand a witness of the required structure as part of the input. However, when the size of a vertex cover is the parameter, this is not a serious demand. Suppose the input parameter is ℓ. We find a 2-approximation of the minimum vertex cover of the input graph G(pp 11,[23]). If the size of the approximate vertex cover is strictly more than 2ℓ, then we have verified that the input parameter does not correspond to a valid vertex cover number of G. Otherwise, the approximate vertex cover is of size 2ℓ and we can use this vertex cover as a witness. Thus, we may assume that we are solving the following problem.

VC-Harmonious Coloring	**Parameter:** $\|X\|$
Input: A graph G, a vertex cover X of G, a non-negative integer k	
Question: Is there a harmonious coloring of G with k colors?	

The idea is to enumerate over all the possible harmonious coloring of $G[X]$ and for each harmonious coloring, verify whether it can be extended to G using a total of k colors. As we will see, the problem of extending harmonious coloring of $G[X]$ to the entire graph is equivalent to that of finding harmonious coloring of the graph such that each color class contains at most one vertex from the vertex cover. We first observe some properties of such a harmonious coloring.

In the remaining section, unless stated otherwise, G is the input graph with vertex cover X of size ℓ and $I = V(G) \setminus X$ is an independent set.

Observation 3 ($\star$). *For any harmonious coloring of G the size of a color class is at most ℓ.*

For each vertex u in I we associate a brand.

Definition 2. *The* brand *of a vertex v in I with respect to X is the set $N(v)$.*

The number of different brands is upper bounded by the number of nonempty subsets of X which is $2^\ell - 1$. For vertices u, v in I if $\mathsf{brand}(u) \cap \mathsf{brand}(v) \neq \emptyset$ then $d_G(u, v) = 2$ and by Observation 1 these two vertices can not belong to the same harmonious color class. For $S \subseteq X$, we define set $I(S) = \{v \in I | \mathsf{brand}(u) = S\}$.

Consider a harmonious coloring $h : V(G) \to [k]$ and two vertices u, v in I, such that $\mathsf{brand}(u) = \mathsf{brand}(v)$. Let $h(u) = i$ and $h(v) = j$. Define a coloring $\widetilde{h}$ on $V(G)$ as $\widetilde{h}(w) = h(w)$ for all w in $V(G) \setminus \{u, v\}$, and $\widetilde{h}(u) = j$ and $\widetilde{h}(v) = i$.

Observation 4 ($\star$). *For a given harmonious coloring h of G, let u, v be two vertices in I such that $\mathsf{brand}(u) = \mathsf{brand}(v)$. If coloring $\widetilde{h}$ is as defined above then $\widetilde{h}$ is also a harmonious coloring of G.*

Thus we can characterize a harmonious color class based on the brand of the vertices which are part of it. Once the brands which make up the color class are fixed, it does not matter which vertex having that brand is chosen for the color class. This leads us to the definition of a type of a potential color class.

Definition 3 (type)**.** *A* type *Z with respect to X is a $\ell+1$ sized tuple where the first entry is subset of X of cardinality at most 1, and each of the remaining ℓ entries is either $\emptyset$ or a distinct* brand *of a vertex in I.*

A type Z can be represented as $(Y; S_1, S_2, \ldots, S_\ell)$ where Y is either an empty set or a singleton set from X. All the entries in this tuple are subsets of X but we distinguish the first entry from the remaining entries. The number of different types is at most $\ell \cdot \binom{2^\ell}{\ell}$, which is at most $\ell \cdot 2^{\ell^2}$. Any color class C which contains at most one vertex from the vertex cover and at most ℓ vertices from the independent set can be labeled with some type.

Definition 4 (Color Class of type Z). *Let h be a harmonious coloring of G such that each color class contains at most one vertex from X, and let $Z = (Y; S_1, S_2, \ldots, S_\ell)$ be a type defined with respect to X. Color class C of h is of type Z if $C \cap X = Y$ and for every $u \in C \cap I$ there exists S_i in type Z such that $\mathsf{brand}(u) = S_i$.*

Not all the types can be used to label a harmonious color class. We define the notion of valid types to filter out such types.

Definition 5 (Valid type). *A type $Z = (Y; S_1, S_2, \ldots, S_\ell)$ is said to be valid if all the sets in the family $\{N[Y], S_1, S_2, \ldots S_\ell\}$ are pairwise disjoint.*

The validity constraints imply that if a vertex set is labeled with a valid type Z, then for any u, v in that set, the minimum distance between u and v is strictly greater than 2. Only the valid types can be used to label harmonious color classes.

Definition 6 (Compatible types). *Two valid types $Z = (Y; S_1, S_2, \ldots, S_\ell)$ and $Z' = (Y'; S'_1, S'_2, \ldots, S'_\ell)$ are said to be compatible with each other if $|Y \cap (S'_1 \cup S'_2 \cup \cdots \cup S'_\ell)| + |Y' \cap (S_1 \cup S_2 \cup \cdots \cup S_\ell)| \leq 1$.*

The compatibility condition of types encodes that the number of edges running across two harmonious color classes is at most 1. Two harmonious color classes C and C' can be of type Z and Z' respectively only if these two types are compatible with each other.

Lemma 3. *Let C and C' are two disjoint sets of $V(G)$ of valid types $Z = (Y; S_1, S_2, \ldots, S_\ell)$ and $Z' = (Y'; S'_1, S'_2, \ldots, S'_\ell)$ respectively. $|E(C, C')| \leq 1$ if and only if Z and Z' are campatible with each other.*

Proof. ($\Rightarrow$) If $|E(C, C')| = 0$ then there is no edge across C' and C and hence $|Y \cap (S'_1 \cup S'_2 \cup \cdots \cup S'_\ell)| = |Y' \cap (S_1 \cup S_2 \cup \cdots \cup S_\ell)| = 0$ making types Z and Z' compatible. Consider the case when $|E(C, C')| = 1$. With out loss of generality, let $x \in C \cap X$ and $z' \in C'$ and xz' is the edge across C and C'. For any u in $C \setminus X$, $E(\{u\}, Y') = \emptyset$ implying $N(u) \cap Y' = \emptyset$ which is equivalent to $|Y' \cap (S_1 \cup S_2 \cup \cdots \cup S_\ell)| = 0$. Since xz' is the only edge across C and C', $|Y \cap (S'_1 \cup S'_2 \cup \cdots \cup S'_\ell)|$ is 0 or 1 depending on whether z' is in X or not. In either case, types Z and Z' are compatible.

($\Leftarrow$) If $|Y \cap (S'_1 \cup S'_2 \cup \cdots \cup S'_\ell)| = |Y' \cap (S_1 \cup S_2 \cup \cdots \cup S_\ell)| = 0$ then there is no edge across C and C' whose one end point is outside vertex cover X. Since Y and Y' has cardinality of at most 1, $|E(C, C')| \leq 1$. So now we are in a case where $|Y \cap (S'_1 \cup S'_2 \cup \cdots \cup S'_\ell)| + |Y' \cap (S_1 \cup S_2 \cup \cdots \cup S_\ell)| = 1$. Without loss of generality, assume that $|Y \cap (S'_1 \cup S'_2 \cup \cdots \cup S'_\ell)| = 1$. This imply that there is an edge whose one end point is in Y and another end point is in $C' \setminus Y'$. Also, $|Y' \cap (S_1 \cup S_2 \cup \cdots \cup S_\ell)| = 0$ implies that there is no edge with one end point incident on Y' and another end point in $C \setminus Y$. The only thing that remains to argue that in this situation $E(Y, Y') = \emptyset$. If this is not the case then $Y \cap N(Y') \neq \emptyset$. But there exists S'_i such that $Y \cap S'_i \neq \emptyset$. Since Y is singleton set, this implies $N(Y') \cap S'_i \neq \emptyset$ which contradicts the fact that type Z' is valid. Hence $E(Y, Y') = \emptyset$ which concludes the proof of $|E(C, C')| \leq 1$. □

For a given graph G and a vertex cover X of G, we construct a set $\mathcal{Z}$ consisting of all types with respect to X which are valid. For every subset $\mathcal{Z}'$ of $\mathcal{Z}$ such that any two types in $\mathcal{Z}'$ are compatible with each other, we construct an instance $\mathcal{J}_{\mathcal{Z}'}$ of Integer Linear Programming as follows.

We define a variable z_i as the number of color class of type Z_i used in the coloring. In the following objective function, we encode the aim of minimizing number of color classes used.

$$\text{minimize} \sum_{i=1}^{|\mathcal{Z}'|} z_i$$

For every $S \subseteq X$ and $j \in [|\mathcal{Z}'|]$ define

$$b_j^S = 1 \text{ if there is } \mathsf{brand}\ S \text{ in } \mathsf{type}\ Z_j; \text{ otherwise } 0$$

There are exactly $|I(S)|$ many vertices of brand S.

$$\sum_{j=1}^{|\mathcal{Z}'|} z_j \cdot b_j^S = |I(S)| \qquad \forall S \subseteq X \tag{1}$$

For every $x \in X$ and $j \in [|\mathcal{Z}'|]$ define

$$c_j^x = 1 \text{ if } \{x\} \text{ is the first entry in } \mathsf{type} Z_j; \text{ otherwise } 0$$

There can be at most one color class which contains vertex x in X.

$$\sum_{j=1}^{|\mathcal{Z}'|} z_j \cdot c_j^x = 1 \qquad \forall x \in X \tag{2}$$

Corollary 2. *An instance $\mathcal{J}_{\mathcal{Z}'}$ can be solved in time $2^{\mathcal{O}(2^{\ell^2} \cdot \ell^3)} n^{\mathcal{O}(1)}$.*

Proof. The number of variables in instance $\mathcal{J}_{\mathcal{Z}'}$ is $|\mathcal{Z}'|$ which is upper bounded by $\ell \cdot 2^{\ell^2}$. The maximum value, any variable z_i can take, is n and the largest value any coefficient in the objective function can take is 1. The coefficients in the constraints are upper bounded by n. The number of constraints is at most $2^\ell + \ell$. By Proposition 1, instance $\mathcal{J}_{\mathcal{Z}'}$ can be solved in time $2^{\mathcal{O}(2^{\ell^2} \cdot \ell^3)} n^{\mathcal{O}(1)}$. □

Recall that for a given graph G and its vertex cover X, $\mathcal{Z}$ is the set of all valid types with respect to X and $\mathcal{Z}'$ is a subset of $\mathcal{Z}$ such that any two types in $\mathcal{Z}'$ are compatible with each other.

Lemma 4 ($\star$). *Given a graph G with a vertex cover X, an integer k, there exists a harmonious coloring of G with at most k colors and each color class contains at most one vertex from X if and only if there exists $\mathcal{Z}' \subseteq \mathcal{Z}$ such that the minimum value for an instance $\mathcal{I}_{\mathcal{Z}'}$ is at most k.*

This leads us to the main theorem of this section.

Theorem 4 ($\star$). Harmonious Coloring*, parameterized by the size of a vertex cover of the input graph, is fixed-parameter tractable.*

While it is an interesting open problem to improve the bound of the FPT algorithm, we show that when the input graph is a forest, the bound can be substantially improved to show the following.

Theorem 5 ($\star$). *Given a forest G, a vertex cover X of size ℓ, we can find the minimum harmonious number, and the corresponding coloring of G in $2^{\mathcal{O}(\ell^2)} n^{\mathcal{O}(1)}$ time.*

The main reason for the improved bound is that the number of brands for vertices in $V(I)$ comes down to at most $2\ell - 1$ (from $2^\ell - 1$). Also, except for ℓ brands, all others have at most one vertex having that brand. Furthermore, we can run through some careful choices and avoid solving the integer linear programming. The details are in the full version of this paper.

5 Exact Algorithm on Split Graphs

As the number of vertices is a trivial upper bound for the harmonious coloring number, a naive algorithm to find the minimum harmonious number runs through all the n^n possible colorings to find the minimum number. It is know that Harmonious Coloring on Split graphs is NP-Complete. In this section, we give an exact algorithm for Harmonious coloring on the class of split graphs improving on this $2^{n \log n}$ bound to $2^n n^{O(1)}$. We make use of a relation between a harmonious coloring of a split graph and a proper coloring of an auxiliary graph to obtain our improved algorithm. We can relate the number of colors required for a harmonious coloring of the graph G with that for a harmonious coloring of its non-trivial component.

Observation 5 ($\star$). *Let G be an input split graph with $E(G) \neq \emptyset$ and let C be a non-trivial component of G. Then $\mathsf{hc}(G) = \mathsf{hc}(C)$.*

Observation 6 ([21]). *For any harmonious coloring h of G and a split-partition (K, I), each vertex in K must be given a distinct color.*

As a corollary to Lemma 1, we obtain the following relation in split graphs.

Corollary 3 (⋆). *Let G be a connected split graph with a split-partition (K, I), and let H be the auxiliary graph defined from G as in the statement of Lemma 1. A coloring function h is a harmonious coloring of G if and only if $(i)h(K) \cap h(I) = \emptyset$, (ii) each vertex of K gets distinct color, and $(iii)h|_I$ is a proper coloring of H.*

Theorem 6. *Given a split graph G, there is an algorithm, running in $2^n n^{\mathcal{O}(1)}$ time, that computes the minimum harmonious coloring of graph G.*

Proof. By Observation 5, we can assume that G is a connected graph. Let (K, I) be a split partition of G. By Observation 6, in any harmonious coloring of G, each vertex of K must get a distinct color. Also, by connectivity, each vertex in $V(I)$ must be adjacent to a vertex in $V(K)$. Hence, in any harmonious coloring of G, the vertices of $V(I)$ must be colored distinctly from the vertices of $V(K)$. From Corollary 3, the minimum proper coloring of the auxiliary graph H gives the minimum harmonious coloring of G extending the coloring of K. Thus, it is enough to find the minimum proper coloring of H, which can be done in time $2^n n^{\mathcal{O}(1)}$ using the algorithm of Björklund et al. [5]. □

We obtain an improved FPT algorithm for split graphs as a corollary.

Corollary 4 (⋆). *Given a split graph G and a non-negative integer k, we can determine whether G has a harmoniously coloring with at most k colors in $2^{\mathcal{O}(k^2)} n^{\mathcal{O}(1)}$ time.*

6 Conclusions

We have shown that the harmonious coloring problem is fixed-parameter tractable when parameterized by the harmonious coloring number or the vertex cover number. While improving the bounds for our FPT algorithms is a natural open problem, we end with the following specific open problems.

- When parameterizing by k, the harmonious coloring number, can the kernel size of $O(k^2)$ be improved?
- When parameterizing by the vertex cover number ℓ, is there a $c^\ell n^{O(1)}$ algorithm, for some constant c, at least on trees?

References

1. Aflaki, A., Akbari, S., Edwards, K., Eskandani, D., Jamaali, M., Ravanbod, H.: On harmonious colouring of trees. Electron. J. Comb. **19**(1), P3 (2012)
2. Appel, K., Haken, W.: Every planar map is four colorable. part i: discharging. Ill. J. Math. **21**(3), 429–490 (1977)
3. Asdre, K., Ioannidou, K., Nikolopoulos, S.D.: The harmonious coloring problem is NP-complete for interval and permutation graphs. Discrete Appl. Math. **155**(17), 2377–2382 (2007)

4. Asdre, K., Nikolopoulos, S.D.: NP-completeness results for some problems on subclasses of bipartite and chordal graphs. Theor. Comput. Sci. **381**(1), 248–259 (2007)
5. Björklund, A., Husfeldt, T., Koivisto, M.: Set partitioning via inclusion-exclusion. SIAM J. Comput. **39**(2), 546–563 (2009)
6. Bodlaender, H.L.: Achromatic number is NP-complete for cographs and interval graphs. Inf. Proc. Lett. **31**(3), 135–138 (1989)
7. Chen, J., Kanj, I.A., Meng, J., Xia, G., Zhang, F.: On the pseudo-achromatic number problem. Theor. Comput. Sci. **410**(810), 818–829 (2009)
8. Cygan, M., Fomin, F.V., Kowalik, $\mathcal{L}$., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: Parameterized Algorithms, vol. 4. Springer, Switzerland (2015)
9. Diestel, R.: Graph Theory. Graduate Texts in Mathematics, vol. 101 (2005)
10. Downey, R.G., Fellows, M.R.: Parameterized Complexity. Springer, New York (1999)
11. Edwards, K.: The harmonious chromatic number of bounded degree trees. Comb. Probab. Comput. **5**(01), 15–28 (1996)
12. Edwards, K.: The harmonious chromatic number and the achromatic number. Surv. Comb. **16**, 13 (1997)
13. Edwards, K., McDiarmid, C.: The complexity of harmonious colouring for trees. Discrete Appl. Math. **57**(2), 133–144 (1995)
14. Hell, P., Miller, D.J.: Graph with given achromatic number. Discrete Math. **16**(3), 195–207 (1976)
15. Hopcroft, J.E., Krishnamoorthy, M.S.: On the harmonious coloring of graphs. SIAM J. Algebraic Discrete Meth. **4**(3), 306–311 (1983)
16. Ioannidou, K., Nikolopoulos, S.D.: Harmonious coloring on subclasses of colinear graphs. In: Rahman, M.S., Fujita, S. (eds.) WALCOM 2010. LNCS, vol. 5942, pp. 136–148. Springer, Heidelberg (2010)
17. Kannan, R.: Minkowski's convex body theorem and integer programming. Math. Oper. Res. **12**(3), 415–440 (1987)
18. Karp, R.M.: Reducibility among combinatorial problems. In: Miller, R.E., Thatcher, J.W., Bohlinger, J.D. (eds.) Complexity of Computer Computations, pp. 85–103. Springer, New York (1972)
19. Lawler, E.L.: A note on the complexity of the chromatic number problem. Inf. Proc. Lett. **5**(3), 66–67 (1976)
20. Lenstra Jr., H.W.: Integer programming with a fixed number of variables. Math. Oper. Res. **8**(4), 538–548 (1983)
21. Miller, Z., Pritikin, D.: The harmonious coloring number of a graph. Discrete Math. **93**(2–3), 211–228 (1991)
22. Panolan, F., Philip, G., Saurabh, S.: B-chromatic number: Beyond np-hardness. In: 10th International Symposium on Parameterized and Exact Computation, IPEC 2015, 16–18 September 2015, Patras, Greece, pp. 389–401 (2015)
23. Papadimitriou, C.H., Steiglitz, K.: Combinatorial Optimization: Algorithms and Complexity. Courier Corporation, New York (1982)