List Coloring in the Absence of Two Subgraphs ⋆

Petr A. Golovach[1] and Daniël Paulusma[2]

[1] Department of Informatics, University of Bergen, Norway,
petr.golovach@ii.uib.no
[2] School of Engineering and Computing Sciences, Durham University,
daniel.paulusma@durham.ac.uk

Abstract. A list assignment of a graph $G = (V, E)$ is a function $\mathcal{L}$ that assigns a list $L(u)$ of so-called admissible colors to each $u \in V$. The List Coloring problem is that of testing whether a given graph $G = (V, E)$ has a coloring c that respects a given list assignment $\mathcal{L}$, i.e., whether G has a mapping $c : V \to \{1, 2, \ldots\}$ such that (i) $c(u) \neq c(v)$ whenever $uv \in E$ and (ii) $c(u) \in L(u)$ for all $u \in V$. If a graph G has no induced subgraph isomorphic to some graph of a pair $\{H_1, H_2\}$, then G is called (H_1, H_2)-free. We completely characterize the complexity of List Coloring for (H_1, H_2)-free graphs.

1 Introduction

Graph coloring involves the labeling of the vertices of some given graph by integers called colors such that no two adjacent vertices receive the same color. The goal is to minimize the number of colors. Graph coloring is one of the most fundamental concepts in both structural and algorithmic graph theory and arises in a vast number of theoretical and practical applications. Many variants are known, and due to its hardness, the graph coloring problem has been well studied for special graph classes such as those defined by one or more forbidden induced subgraphs. We consider a more general version of graph coloring called list coloring and classify the complexity of this problem for graphs characterized by two forbidden induced subgraphs. Kratsch and Schweitzer [22] and Lozin [23] performed a similar study as ours for the problems graph isomorphism and dominating set, respectively. Before we summarize related coloring results and explain our new results, we first state the necessary terminology. For a more general overview of the area we refer to the surveys of Randerath and Schiermeyer [29] and Tuza [32], and to the book by Jensen and Toft [26].

1.1 Terminology

We only consider finite undirected graphs with no multiple edges and self-loops. A *coloring* of a graph $G = (V, E)$ is a mapping $c : V \to \{1, 2, \ldots\}$ such that $c(u) \neq c(v)$ whenever $uv \in E$. We call $c(u)$ the *color* of u. A *k-coloring* of G is

⋆ This paper was supported by EPSRC (EP/G043434/1) and ERC (267959).

a coloring c of G with $1 \leq c(u) \leq k$ for all $u \in V$. The Coloring problem is that of testing whether a given graph admits a k-coloring for some given integer k. If k is *fixed*, i.e., not part of the input, then we denote the problem as k-Coloring. A *list assignment* of a graph $G = (V, E)$ is a function $\mathcal{L}$ that assigns a list $L(u)$ of so-called *admissible* colors to each $u \in V$. If $L(u) \subseteq \{1, \ldots, k\}$ for each $u \in V$, then $\mathcal{L}$ is also called a *k-list assignment*. We say that a coloring $c\colon V \to \{1, 2, \ldots\}$ *respects* $\mathcal{L}$ if $c(u) \in L(u)$ for all $u \in V$. The List Coloring problem is that of testing whether a given graph has a coloring that respects some given list assignment. For a fixed integer k, the List k-Coloring problem has as input a graph G with a k-list assignment $\mathcal{L}$ and asks whether G has a coloring that respects $\mathcal{L}$. The *size* of a list assignment $\mathcal{L}$ is the maximum list size $|L(u)|$ over all vertices $u \in V$. For a fixed integer ℓ, the ℓ-List Coloring problem has as input a graph G with a list assignment $\mathcal{L}$ of size at most ℓ and asks whether G has a coloring that respects $\mathcal{L}$. Note that k-Coloring can be viewed as a special case of List k-Coloring by choosing $L(u) = \{1, \ldots, k\}$ for all vertices u of the input graph, whereas List k-Coloring is readily seen to be a special case of k-List Coloring.

For a subset $S \subseteq V(G)$, we let $G[S]$ denote the *induced* subgraph of G, i.e., the graph with vertex set S and edge set $\{uv \in E(G) \mid u, v \in S\}$. For a graph F, we write $F \subseteq_i G$ to denote that F is an induced subgraph of G. Let G be a graph and $\{H_1, \ldots, H_p\}$ be a set of graphs. We say that G is *$(H_1, \ldots, H_p)$-free* if G has no induced subgraph isomorphic to a graph in $\{H_1, \ldots, H_p\}$; if $p = 1$, we may write H_1-free instead of (H_1)-free.

The *complement* of a graph $G = (V, E)$ denoted by $\overline{G}$ has vertex set V and an edge between two distinct vertices if and only if these vertices are not adjacent in G. The *union* of two graphs G and H is the graph with vertex set $V(G) \cup V(H)$ and edge set $E(G) \cup E(H)$. Note that G and H may share some vertices. If $V(G) \cap V(H) = \emptyset$, then we speak of the *disjoint union* of G and H denoted by $G + H$. We denote the disjoint union of r copies of G by rG. The graphs C_r, P_r and K_r denote the cycle, path, and complete graph on r vertices, respectively. The graph $K_{r,s}$ denotes the complete bipartite graph with partition classes of size r and s, respectively. The graph K_4^- denotes the *diamond*, which is the complete graph on four vertices minus an edge. The *line graph* of a graph G with edges $e_1, \ldots, e_p$ is the graph with vertices $u_1, \ldots, u_p$ such that there is an edge between any two vertices u_i and u_j if and only if e_i and e_j share an end-vertex in G.

1.2 Related Work

Král' et. al. [20] completely determined the computational complexity of Coloring for graph classes characterized by one forbidden induced subgraph. By combining a number of known results, Golovach, Paulusma and Song [13] obtained similar dichotomy results for the problems List Coloring and k-List Coloring, whereas the complexity classifications of the problems List k-Coloring and k-Coloring are still open (see, e.g., [14] for a survey).

Theorem 1. *Let H be a fixed graph. Then the following three statements hold:*

(i) Coloring *is polynomial-time solvable for H-free graphs if H is an induced subgraph of P_4 or of $P_1 + P_3$; otherwise it is* NP*-complete for H-free graphs.*

(ii) List Coloring *is polynomial-time solvable for H-free graphs if H is an induced subgraph of P_3; otherwise it is* NP*-complete for H-free graphs.*

*(iii) For all $\ell \leq 2$, ℓ-*List Coloring *is polynomial-time solvable. For all $\ell \geq 3$, ℓ-*List Coloring *is polynomial-time solvable for H-free graphs if H is an induced subgraph of P_3; otherwise it is* NP*-complete for H-free graphs.*

When we forbid *two* induced subgraphs the situation becomes less clear for the Coloring problem, and only partial results are known. We summarize these results in the following theorem. Here, C_3^+ denotes the graph with vertices a, b, c, d and edges ab, ac, ad, bc, whereas F_5 denote the 5-vertex fan also called the gem, which is the graph with vertices a, b, c, d, e and edges $ab, bc, cd, ea, eb, ec, ed$.

Theorem 2. *Let H_1 and H_2 be two fixed graphs. Then the following holds:*

(i) Coloring *is* NP*-complete for (H_1, H_2)-free graphs if*

1. *$H_1 \supseteq_i C_r$ for some $r \geq 3$ and $H_2 \supseteq_i C_s$ for some $s \geq 3$*
2. *$H_1 \supseteq_i K_{1,3}$ and $H_2 \supseteq_i K_{1,3}$*
3. *H_1 and H_2 contain a spanning subgraph of $2P_2$ as an induced subgraph*
4. *$H_1 \supseteq_i C_3$ and $H_2 \supseteq_i K_{1,r}$ for some $r \geq 5$*
5. *$H_1 \supseteq_i C_3$ and $H_2 \supseteq_i P_{164}$*
6. *$H_1 \supseteq_i C_r$ for $r \geq 4$ and $H_2 \supseteq_i K_{1,3}$*
7. *$H_1 \supseteq_i C_r$ for $r \geq 5$ and H_2 contains a spanning subgraph of $2P_2$ as an induced subgraph*
8. *$H_1 \supseteq_i K_4$ or $H_1 \supseteq_i K_4^-$, and $H_2 \supseteq_i K_{1,3}$*
9. *$H_1 \supseteq_i C_r + P_1$ for $3 \leq r \leq 4$ or $H_1 \supseteq_i \overline{C_r}$ for $r \geq 6$, and H_2 contains a spanning subgraph of $2P_2$ as an induced subgraph.*

(ii) Coloring *is polynomial-time solvable for (H_1, H_2)-free graphs if*

1. *H_1 or H_2 is an induced subgraph of $P_1 + P_3$ or of P_4*
2. *$H_1 \subseteq_i C_3 + P_1$ and $H_2 \subseteq_i K_{1,3}$*
3. *$H_1 \subseteq_i C_3^+$ and $H_2 \neq K_{1,5}$ is a forest on at most six vertices*
4. *$H_1 \subseteq_i C_3^+$, and $H_2 \subseteq_i sP_2$ or $H_2 \subseteq_i sP_1 + P_5$ for $s \geq 1$*
5. *$H_1 = K_r$ for $r \geq 4$, and $H_2 \subseteq_i sP_2$ or $H_2 \subseteq_i sP_1 + P_5$ for $s \geq 1$*
6. *$H_1 \subseteq_i F_5$, and $H_2 \subseteq_i P_1 + P_4$ or $H_2 \subseteq_i P_5$*
7. *$H_1 \subseteq_i \overline{P_5}$, and $H_2 \subseteq_i P_1 + P_4$ or $H_2 \subseteq_i 2P_2$.*

Proof. Král' et al. [20] proved Cases (i):1–4, 6–8. Golovach et al. [12] proved that 4-Coloring is NP-complete for (C_3, P_{164})-free graphs; this shows Case (i):5. Case (i):9 follows from the following result by Schindl [31]. For $1 \leq i \leq j \leq k$, let $S_{h,i,j}$ be the tree with only one vertex x of degree 3 that has exactly three leaves, which are of distance h, i and j to x, respectively. Let $A_{h,i,j}$ be the line graph of $S_{h,i,j}$. Then, for a finite set of graphs $\{H_1, \ldots, H_p\}$, Coloring is

NP-complete for $(H_1, \ldots, H_p)$-free graphs if the complement of each H_i has a connected component isomorphic neither to any graph $A_{i,j,k}$ nor to any path P_r.

Case (ii):1 follows from Theorem 1 (i). Because Coloring can be solved in polynomial time on graphs of bounded clique-width [19], and $(C_3+P_1, K_{1,3})$-free graphs [2], (F_5, P_1+P_4)-free graphs [4], (F_5, P_5)-free graphs [3] and $(\overline{P_5}, P_1+P_4)$-free graphs [3] have bounded clique-width, Cases (ii):2,6–7 hold after observing in addition that $(\overline{P_5}, 2P_2)$-free graphs are b-perfect and Coloring is polynomial-time solvable on b-perfect graphs [16]. Gyárfás [15] showed that for all $r, t \geq 1$, (K_r, P_t)-free graphs can be colored with at most $(t-1)^{r-2}$ colors. Hence, Coloring is polynomial-time solvable on (K_r, F)-free graphs for some linear forest F if k-Coloring is polynomial-time solvable on F-free graphs for all $k \geq 1$. The latter is true for $F = sP_1 + P_5$ [7] and $F = sP_2$ (see e.g. [9]). This shows Case (ii):5, whereas we obtain Case (ii):4 by using the same arguments together with a result of Král' et al. [20], who showed that for any fixed graph H_2, Coloring is polynomial-time solvable on (C_3, H_2)-free graphs if and only if it is so for (C_3^+, H_2)-free graphs. Case (ii):3 is showed by combining the latter result with corresponding results from Dabrowski et al. [9] for (C_3, H_2)-free graphs obtained by combining a number of new results with some known results [5, 6, 24, 27, 28]. $\square$

1.3 Our Contribution

We completely classify the complexity of List Coloring and ℓ-List Coloring for (H_1, H_2)-free graphs. For the latter problem we may assume that $\ell \geq 3$ due to Theorem 1 (iii).

Theorem 3. *Let H_1 and H_2 be two fixed graphs. Then* List Coloring *is polynomial-time solvable for (H_1, H_2)-free graphs in the following cases:*

1. *$H_1 \subseteq_i P_3$ or $H_2 \subseteq_i P_3$*
2. *$H_1 \subseteq_i C_3$ and $H_2 \subseteq_i K_{1,3}$*
3. *$H_1 = K_r$ for some $r \geq 3$ and $H_2 = sP_1$ for some $s \geq 3$.*

In all other cases, even 3-List Coloring *is* NP*-complete for (H_1, H_2)-free graphs.*

We note that the classification in Theorem 3 differs from the partial classification in Theorem 2. For instance, Coloring is polynomial-time solvable on $(C_3, K_{1,4})$-free graphs, whereas 3-List Coloring is NP-complete for this graph class. We prove Theorem 3 in Section 2, whereas Section 3 contains some concluding remarks. There, amongst others, we give a complete classification of the computational complexity of List Coloring and List 3-Coloring when a set of (not necessarily induced) subgraphs is forbidden.

2 The Classification

A graph G is a *split* graph if its vertices can be partitioned into a clique and an independent set; if every vertex in the independent set is adjacent to every vertex in the clique, then G is a *complete* split graph. The graph $K_n - M$ denotes a *complete graph minus a matching* which is obtained from a complete graph K_n after removing the edges of some matching M. Equivalently, a graph G is a complete graph minus a matching if and only if G is $(3P_1, P_1 + P_2)$-free [13]. The complement of a bipartite graph is called a *cobipartite* graph. Let G be a connected bipartite graph with partition classes A and B. Then we call $\overline{G}$ a *matching-separated* cobipartite graph if the edges of $\overline{G}$ that are between vertices from A and B form a matching in $\overline{G}$. The *girth* of a graph G is the length of a shortest induced cycle in G.

For showing the NP-complete cases in Theorem 3 we consider a number of special graph classes in the following three lemmas.

Lemma 1. 3-List Coloring *is* NP-*complete for:*

(i) *complete bipartite graphs*
(ii) *complete split graphs*
(iii) *(non-disjoint) unions of two complete graphs*
(iv) *complete graphs minus a matching*

Proof. The proof of Theorem 4.5 in the paper by Jansen and Scheffler [18] is to show that List Coloring is NP-complete on P_4-free graphs but in fact shows that 3-List Coloring is NP-complete for complete bipartite graphs. This shows (i). In the proof of Theorem 2 in the paper by Golovach and Heggernes [10] a different NP-hardness reduction is given for showing that 3-List Coloring is NP-complete for complete bipartite graphs. In this reduction a complete bipartite graph is constructed with a list assignment that has the following property: all the lists of admissible colors of the vertices for one bipartition class are mutually disjoint. Hence, by adding all possible edges between the vertices in this class, one proves that 3-List Coloring is NP-complete for complete split graphs. This shows (ii). Golovach et al. [13] showed (iii). The proof of Theorem 11 in the paper by Jansen [17] is to show that List Coloring is NP-complete for unions of two complete graphs that are not disjoint unions, but in fact shows that 3-List Coloring is NP-complete for these graphs. This shows (iv). □

Lemma 2. 3-List Coloring *is* NP-*complete for matching-separated cobipartite graphs.*

Proof. NP-membership is clear. To show NP-hardness we reduce from Satisfiability. It is known (see e.g. [8]) that this problem remains NP-complete even if each clause contains either 2 or 3 literals and each variable is used in at most 3 clauses. Consider an instance of Satisfiability with n variables $x_1, \ldots, x_n$ and m clauses $C_1, \ldots, C_m$ that satisfies these two additional conditions. Let $\phi = C_1 \wedge \ldots \wedge C_m$. We construct a graph G with a list assignment $\mathcal{L}$ as follows (see Fig. 1).

- For each $i \in \{1, \ldots, n\}$, add six vertices $x_i^1, x_i^2, x_i^3, y_i^1, y_i^2, y_i^3$, introduce six new colors $i_1, i_2, i_3, i_1', i_2', i_3'$, assign lists of admissible colors $\{i_1, i_1'\}$, $\{i_2, i_2'\}$, $\{i_3, i_3'\}$ to x_i^1, x_i^2, x_i^3, respectively, and $\{i_1, i_2'\}, \{i_2, i_3'\}, \{i_3, i_1'\}$ to y_i^1, y_i^2, y_i^3, respectively.

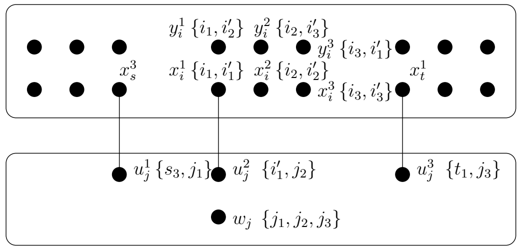

Fig. 1. An example of a graph G with a clause vertex $C_j = \overline{x}_s \vee x_i \vee \overline{x}_t$, where x_s, x_i, x_t occur for the third, first and first time in ϕ, respectively.

- Add edges between all vertices x_i^h, y_i^h to obtain a clique with $6n$ vertices.
- For $j = 1, \ldots, m$, add four vertices u_j^1, u_j^2, u_j^3, w_j, introduce three new colors j_1, j_2, j_3, assign the list of admissible colors $\{j_1, j_2, j_3\}$ to w_j, and if C_j contains exactly two literals, then assign the list $\{j_3\}$ to u_j^3.
- Add edges between all vertices u_j^h, w_j to obtain a clique with $4m$ vertices.
- For $j = 1, \ldots, m$, consider the clause C_j and suppose that $C_j = z_1 \vee z_2$ or $C_j = z_1 \vee z_2 \vee z_3$. For $h = 1, 2, 3$ do as follows:
  - if $z_h = x_i$ is the p-th occurrence of the variable x_i in ϕ, then add the edge $u_j^h x_i^p$ and assign the list of colors $\{i'_p, j_h\}$ to u_j^h;
  - if $z_h = \overline{x}_i$ is the p-th occurrence of the variable x_i in ϕ, then add the edge $u_j^h x_i^p$ and assign the list of colors $\{i_p, j_h\}$ to u_j^h.

  Notice that all the colors $i_1, i_2, i_3, i'_1, i'_2, i'_3,$ j_1, j_2, j_3 are distinct. From its construction, G is readily seen to be a matching-separated cobipartite graph.

We claim that ϕ has a satisfying truth assignment if and only if G has a coloring that respects $\mathcal{L}$. First suppose that ϕ has a satisfying truth assignment. For $i = 1, \ldots, n$, we give the vertices x_i^1, x_i^2, x_i^3 colors i_1, i_2, i_3, respectively, and the vertices y_i^1, y_i^2, y_i^3 colors i'_2, i'_3, i'_1 respectively, if $x_i = true$, and we give x_i^1, x_i^2, x_i^3 colors i'_1, i'_2, i'_3, respectively, and y_i^1, y_i^2, y_i^3 colors i_1, i_2, i_3 respectively, if $x_i = false$. For $j = 1, \ldots, m$, consider the clause C_j and suppose that $C_j = z_1 \vee z_2$ or $C_j = z_1 \vee z_2 \vee z_3$. Note that if C_j contains exactly two literals, then u_j^3 is colored by j_3. The clause C_j contains a literal $z_h = true$. Assume first that $z_h = x_i$ and that z_h is the p-th occurrence of the variable x_i in ϕ. Recall that u_j^h has list of admissible colors $\{i'_p, j_h\}$ and that u_j^h is adjacent to x_i^p colored by i_p. Hence, we color u_j^h by i'_p, w_j by j_h, and for $s \in \{1, 2, 3\} \setminus \{h\}$, we color u_j^s by j_s. Assume now that $z_h = \overline{x}_i$ and that z_h is the p-th occurrence of the variable x_i in ϕ. Symmetrically, we color u_j^h by i_p, w_j by j_h, and for $s \in \{1, 2, 3\} \setminus \{h\}$, we color u_j^s by j_s. We observe that for any distinct $i, i' \in \{1, \ldots, n\}$, the lists of admissible colors of $x_i^1, x_i^2, x_i^3, y_i^1, y_i^2, y_i^3$ do not share any color with the lists of $x_{i'}^1, x_{i'}^2, x_{i'}^3, y_{i'}^1, y_{i'}^2, y_{i'}^3$. Also for any distinct $j, j' \in \{1, \ldots, m\}$, the lists of colors of u_j^1, u_j^2, u_j^3, w_j do not share any color with he lists of $u_{j'}^1, u_{j'}^2, u_{j'}^3, w_{j'}$. Hence we obtained a coloring of G that respects $\mathcal{L}$.

Now suppose that c is a coloring of G that respects $\mathcal{L}$. We need the following claim that holds for all $1 \leq i \leq n$:

either $c(x_i^1) = i_1$, $c(x_i^2) = i_2$, $c(x_i^3) = i_3$ or $c(x_i^1) = i'_1$, $c(x_i^2) = i'_2$, $c(x_i^3) = i'_3$.

In order to see this claim, first assume that $c(x_i^1) = i_1$. Then $c(y_i^1) = i'_2$, $c(x_i^2) = i_2$, $c(y_i^2) = i'_3$, and $c(x_i^3) = i_3$. Symmetrically, if $c(x_i^1) = i'_1$, then $c(y_i^3) = i_3$, $c(x_i^3) = i'_3$, $c(y_i^2) = i_2$, and $c(x_i^2) = i'_2$. Hence, the claim holds, and we can do as follows. For $i = 1, \ldots, n$, we let $x_i = true$ if $c(x_i^1) = i_1$, $c(x_i^2) = i_2$, $c(x_i^3) = i_3$, and $x_i = false$ if $c(x_i^1) = i'_1$, $c(x_i^2) = i'_2$, $c(x_i^3) = i'_3$. We claim that this truth assignment satisfies ϕ. For $j \in \{1, \ldots, m\}$, consider the clause C_j and suppose that $C_j = z_1 \vee z_2$ or $C_j = z_1 \vee z_2 \vee z_3$. Recall that if C_j contains exactly two literals, then $c(u_j^3) = j_3$. We also observe that there is an index $h \in \{1, 2, 3\}$ such that $c(u_j^h) \neq j_h$ as otherwise it would be impossible to color w_j. Hence, if z_h is the p-th occurrence of the variable x_i in ϕ, then $c(u_j^h) = i'_p$ if $z_h = x_i$ and $c(u_j^h) = i_p$ if $z_h = \overline{x}_i$. If $c(u_j^h) = i'_p$, then $c(u_j^h) \neq c(x_i^p) = i_p$, and $x_i = true$. Otherwise, if $c(u_j^h) = i_p$, then $c(u_j^h) \neq c(x_i^p) = i'_p$, and $x_i = false$. In both cases C_j is satisfied. We therefore find that ϕ is satisfied. This completes the proof of Lemma 2. □

Lemma 3. List 3-Coloring *is* NP*-compete for graphs of maximum degree at most* 3 *with girth at least* g*, and in which any two vertices of degree* 3 *are of distance at least* g *from each other, for any fixed constant* $g \geq 3$.

Proof. NP-membership is clear. To show NP-hardness we reduce from a variant of Not-All-Equal Satisfiability with positive literals only. This problem is NP-complete [30] and defined as follows. Given a set $X = \{x_1, x_2, ..., x_n\}$ of logical variables, and a set $\mathcal{C} = \{C_1, C_2, ..., C_m\}$ of clauses over X in which all literals are positive, does there exist a truth assignment for X such that each clause contains at least one true literal and at least one false literal? The variant we consider takes as input an instance $(\mathcal{C}, X)$ of Not-All-Equal Satisfiability with positively literals only that has two additional properties. First, each C_i contains either two or three literals. Second, each literal occurs in at most three different clauses. One can prove that this variant is NP-complete by a reduction from the original problem via a well-known folklore trick (see e.g. [13]).

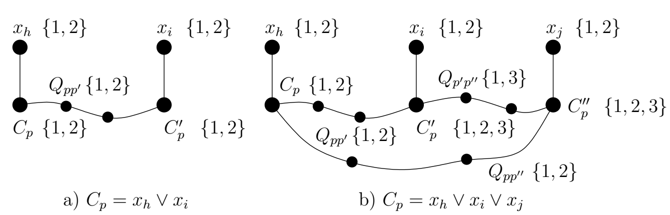

Fig. 2. The construction of G and $\mathcal{L}$ for $g = 3$.

From an instance $(\mathcal{C}, X)$ as defined above, we construct a graph G and a list assignment $\mathcal{L}$ as follows. For each literal x_i we introduce a vertex that we denote by x_i as well. We define $L(x_i) = \{1, 2\}$. For each clause C_p with two literals, we fix an ordering of its literals, say x_h, x_i. We then introduce two vertices C_p, C'_p and add the edges $C_p x_h$ and $C'_p x_i$. We let C_p and C'_p be the end-vertices of a path $Q_{pp'}$ of odd length at least g, whose inner vertices are new vertices. We assign the list $\{1,2\}$ to each vertex of $Q_{pp'}$. See Fig. 2 a). For each clause C_p with three literals, we fix an ordering of its literals, say x_h, x_i, x_j. We then introduce three vertices C_p, C'_p, C''_p and add edges $C_p x_h$, $C'_p x_i$, $C''_p x_j$. We define $L(C_p) = \{1,2\}$ and $L(C'_p) = L(C''_p) = \{1,2,3\}$. We define paths $Q_{pp'}$, $Q_{pp''}$ and $Q_{p'p''}$, each with new inner vertices and of odd length at least g, that go from C_p to C'_p, from C_p to C''_p, and from C'_p to C''_p, respectively. We assign the list $\{1,2\}$ to each inner vertex of $Q_{pp'}$ and to each inner vertex of $Q_{pp''}$, whereas we assign the list $\{1,3\}$ to each inner vertex of $Q_{p'p''}$. See Fig. 2 b). This completes our construction of G and $\mathcal{L}$. Because each clause contains at most three literals and each literal occurs in at most three clauses, G has maximum degree at most 3. By construction, G has girth at least g and any two vertices of degree 3 have distance at least g from each other. We claim that X has a truth assignment such that each clause contains at least one true literal and at least one false literal if and only if G has a coloring that respects $\mathcal{L}$.

First suppose that X has a truth assignment such that each clause contains at least one true literal and at least one false literal. We assign color 1 to every true literal and color 2 to every false literal. Suppose that C_p is a clause containing exactly two literals ordered as x_h, x_i Then, by our assumption, one of them is true and the other one is false. Suppose that x_h is true and x_i is false. Then we give C_p color 2 and C'_p color 1. Because the path $Q_{pp'}$ has odd length, we can alternate between the colors 1 and 2 for the inner vertices of $Q_{pp'}$. If x_h is false and x_i is true, we act in a similar way. Suppose that C_p is a clause containing three literals ordered as x_h, x_i, x_j. By assumption, at least one of the vertices x_h, x_i, x_j received color 1, and at least one of them received color 2. This leaves us with six possible cases. If x_h, x_i, x_j have colors $1,1,2$, then we give C_p, C'_p, C''_p colors 2,3,1, respectively. If x_h, x_i, x_j have colors $1,2,1$, then we give C_p, C'_p, C''_p colors 2,1,3, respectively. If x_h, x_i, x_j have colors $2,1,1$, then we give C_p, C'_p, C''_p colors 1,3,2, respectively. If x_h, x_i, x_j have colors $2,2,1$, then we give C_p, C'_p, C''_p colors 1,3,2, respectively. If x_h, x_i, x_j have colors $2,1,2$, then we give C_p, C'_p, C''_p colors 1,2,3, respectively. If x_h, x_i, x_j have colors $1,2,2$, then we give C_p, C'_p, C''_p colors 2,3,1, respectively. What is left to do is to color the inner vertices of the paths $Q_{pp'}$, $Q_{pp''}$, $Q_{p'p''}$. For the inner vertices of the first two paths we alternate between colors 1 and 2, whereas we alternate between colors 1 and 3 for the inner vertices of the last path. Because we ensured that in all six cases the vertices C_p, C'_p and C''_p received distinct colors and the length of the paths is odd, we can do this. Hence, we obtained a coloring of G that respects $\mathcal{L}$.

Now suppose that G has a coloring that respects $\mathcal{L}$. Then every literal vertex has either color 1 or color 2. In the first case we make the corresponding literal true, and in the second case we make it false. We claim that in this way we obtained a truth assignment of X such that each clause contains at least one true literal and at least one false literal. In order to obtain a contradiction suppose that C_p is a clause, all literals of which are either true or false. First suppose that all its literals are true, i.e., they all received color 1. If C_p contains exactly two literals, then both C_p and C'_p received color 2, which is not possible. If C_p contains three literals, then C_p received color 2. Consequently, the colors of the inner vertices of the path $Q_{pp'}$ are forced. Because $Q_{pp'}$ has odd length, this means that the neighbor of C'_p that is on $Q_{pp'}$ received color 2. Then, because C'_p is adjacent to a literal vertex with color 1, we find that C'_p must have received color 3. However, following the same arguments, we now find that the three neighbors of $C_{p''}$ have colors 1,2,3, respectively. This is not possible. If all literals of C_p are false, we use the same arguments to obtain the same contradiction. Hence, such a clause C_p does not exist. This completes the proof of Lemma 3. □

Note that Lemmas 1 and 2 claim NP-completeness for 3-List Coloring on some special graph classes, whereas Lemma 3 claims this for List 3-Coloring, which is the more restricted version of List Coloring where only three distinct colors may be used in total as admissible colors in the lists of a list assignment.

We are now ready to prove Theorem 3.

Proof (of Theorem 3). We first show the polynomial-time solvable cases. Case 1 follows from Theorem 1 (ii). Any $(C_3, K_{1,3})$-free graph has maximum degree at most 2. Kratochvíl and Tuza [21] showed that List Coloring is polynomial-time solvable on graphs of maximum degree 2. This proves Case 2. By Ramsey's Theorem, every (K_r, sP_1)-free graph contains at most $\gamma(r, s)$ vertices for some constant $\gamma(r, s)$. Hence, we can decide in constant time whether such a graph has a coloring that respects some given list assignment. This proves Case 3.

Suppose that Cases 1–3 are not applicable. If both H_1 and H_2 contain a cycle, then NP-completeness of 3-List Coloring follows from Theorem 2 (i):1. Suppose that one of the graphs, say H_1, contains a cycle, whereas H_2 contains no cycle, i.e., is a forest.

First suppose that H_1 contains an induced C_r for some $r \geq 4$. Because H_2 is not an induced subgraph of P_3, we find that H_2 contains an induced $P_1 + P_2$ or an induced $3P_1$. If H_2 contains an induced $P_1 + P_2$, then every complete split graph is (H_1, H_2)-free. Hence NP-completeness of 3-List Coloring follows from Lemma 1 (ii). If H_2 contains an induced $3P_1$, then every union of two complete graphs is (H_1, H_2)-free. Hence NP-completeness of 3-List Coloring follows from Lemma 1 (iii).

Now suppose that H_1 contains no induced C_r for some $r \geq 4$, but suppose that it does contain C_3. If H_2 contains an induced $P_1 + P_2$, then every complete bipartite graph is (H_1, H_2)-free. Hence NP-completeness of 3-List Coloring follows from Lemma 1 (i). If H_2 contains an induced $K_{1,r}$ for some $r \geq 4$, then every graph of maximum degree at most 3 and of girth at least 4 is (H_1, H_2)-free. Hence, NP-completeness of 3-List Coloring follows from Lemma 3 after choosing $g = 4$. Suppose that H_2 contains neither an induced $P_1 + P_2$ nor an induced $K_{1,r}$ for some $r \geq 4$. Recall that H_2 is a forest that is not an induced subgraph of P_3. Then $H_2 = sP_1$ for some $s \geq 3$ or $H_2 = K_{1,3}$.

First suppose that $H_2 = sP_1$ for some $s \geq 3$. If H_1 is not a complete graph minus a matching, then every complete graph minus a matching is (H_1, H_2)-

free. Hence NP-completeness of 3-List Coloring follows from Lemma 1 (iv). If H_1 is not a non-disjoint union of two complete graphs, then every non-disjoint union of two complete graphs is (H_1, H_2)-free. Hence NP-completeness of 3-List Coloring follows from Lemma 1 (iii). Now assume that H_1 is a complete graph minus a matching and also the non-disjoint union of two complete graphs. Then either H_1 is a complete graph or a complete graph minus an edge. However, H_1 is not a complete graph by assumption (as otherwise we would end up in Case 3 again). Hence H_1 is a complete graph minus an edge. Because H_1 contains C_3, this means that H_1 contains an induced K_4^-. However, then every matching-separated cobipartite graph is (H_1, H_2)-free. Hence NP-completeness of 3-List Coloring follows from Lemma 2.

Now suppose that $H_2 = K_{1,3}$. By repeating the arguments of the previous case, in which $H_2 = sP_1$ for some $s \geq 3$, we obtain NP-completeness of 3-List Coloring or find that H_1 is a complete graph or a complete graph minus an edge. If H_1 is a complete graph, then $H_1 \neq C_3$ by assumption (as otherwise we would end up in Case 2 again). This means that H_1 contains an induced K_4. If H_1 is a complete graph minus an edge, then H_1 contains an induced K_4^- as H_1 already contains the graph C_3. Hence, in both cases, every $(K_4, K_4^-, K_{1,3})$-free graph is (H_1, H_2)-free. Observation 3 in the paper of Král' et al. [20] tells us that Coloring is NP-complete for $(K_4, K_4^-, K_{1,3})$-free graphs. However, its proof shows in fact that 3-Coloring is NP-compete for this graph class. Hence, NP-completeness of 3-List Coloring follows.

Finally we consider the case when H_1 and H_2 contain no cycles, i.e., are both forests. Because neither of them is an induced subgraph of P_3, each of them contains an induced $3P_1$ or an induced $P_1 + P_2$. Recall that a graph is a complete graph minus a matching if and only if it is $(3P_1, P_2)$-free. Hence, any complete graph minus a matching is (H_1, H_2)-free. Then NP-completeness of 3-List Coloring follows from Lemma 1 (iv). This completes the proof of Theorem 3. □

3 Conclusion

We completely classified the complexity of List Coloring and ℓ-List Coloring for (H_1, H_2)-free graphs. The next step would be to classify these two problems for $\mathcal{H}$-free graphs, where $\mathcal{H}$ is an arbitrary finite set of graphs. However, even the case with three forbidden induced subgraphs is not clear. This is in stark contrast to the situation when we forbid subgraphs that may not necessarily be induced. For a set of graphs $\{H_1, \ldots, H_p\}$, we say that a graph G is *strongly* $(H_1, \ldots, H_p)$*-free* if G has no subgraph isomorphic to a graph in $\{H_1, \ldots, H_p\}$. For such graphs we can show the following result.

Theorem 4. *Let $\{H_1, \ldots, H_p\}$ be a finite set of graphs. Then* List Coloring *is polynomial-time solvable for strongly $(H_1, \ldots, H_p)$-free graphs if there exists a graph H_i that is a forest of maximum degree at most* 3*, every connected component of which has at most one vertex of degree* 3*. In all other cases, even* List 3-Coloring *is* NP*-complete for $(H_1, \ldots, H_p)$-free graphs.*

Proof. First suppose there exists a graph H_i that is a forest of maximum degree at most 3, in which every connected component contains at most one vertex of degree 3. Because H_i has maximum degree at most 3, every connected component of H_i is either a path or a subdivided claw. As such, H_i is not a subgraph of a graph G if and only if H is not a minor of G. In that case G has path-width at most $|V(H)| - 2$ [1]. Then the path-width, and hence, the treewidth of G is bounded, as H is fixed. Because List Coloring is polynomial-time solvable for graphs of bounded treewidth [18], we find that List Coloring is polynomial-time solvable for strongly H_i-free graphs, and consequently, for strongly $(H_1, \ldots, H_p)$-free graphs. Now suppose that we do not have such a graph H_i. Then every H_i contains either an induced cycle or is a forest with a vertex of degree at least 4 or is forest that contains a connected component with two vertices of degree 3. Then NP-completeness of List 3-Coloring follows from Lemma 3 after choosing the constant g sufficiently large. $\square$

We note that a classification for Coloring and k-Coloring similar to the one in Theorem 4 for List Coloring and List 3-Coloring is not known even if only one (not necessarily induced) subgraph is forbidden; see Golovach et al. [11] for partial results in this direction.

Another interesting problem, which is still open, is the following. It is not difficult to see that k-Coloring is NP-complete for graphs of diameter d for all pairs (k, d) with $k \geq 3$ and $d \geq 2$ except when $(k, d) \in \{(3, 2), (3, 3)\}$. Recently, Mertzios and Spirakis [25] solved one of the two remaining cases by showing that 3-Coloring is NP-complete even for triangle-free graphs $G = (V, E)$ of diameter 3, radius 2 and minimum degree $\delta = \theta(|V|^\epsilon)$ for every $0 \leq \epsilon \leq 1$. This immediately implies that List 3-Coloring is NP-complete for graphs of diameter 3. What is the computational complexity of List 3-Coloring for graphs of diameter 2?

References

1. D. Bienstock, N. Robertson, P. D. Seymour, and R. Thomas, Quickly excluding a forest, J. Comb. Theory, Ser. B 52 (1991) 274–283.
2. A. Brandstädt, J. Engelfriet, H.-O. Le, and V.V. Lozin: Clique-Width for 4-Vertex Forbidden Subgraphs, Theory Comput. Syst. 39 (2006) 561–590.
3. A. Brandstädt and D. Kratsch, On the structure of (P_5, gem)-free graphs, Discrete Applied Mathematics 145 (2005) 155–166.
4. A. Brandstädt, H.-O. Le, R. Mosca, Gem- and co-gem-free graphs have bounded clique-width, Internat. J. Found. Comput Sci. 15 (2004), 163–185.
5. H.J. Broersma, P.A. Golovach, D. Paulusma and J. Song, Determining the chromatic number of triangle-free $2P_3$-free graphs in polynomial time, Theoretical Computer Science 423 (2012) 1–10.
6. H.J. Broersma, P.A. Golovach, D. Paulusma and J. Song, Updating the complexity status of coloring graphs without a fixed induced linear forest, Theoretical Computer Science 414 (2012) 9–19.
7. J.F. Couturier, P.A. Golovach, D. Kratsch and D. Paulusma, List coloring in the absence of a linear forest, Proc. WG 2011, LNCS 6986 (2011) 119–130.

8. E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis, The complexity of multiterminal cuts, SIAM J. Comput. 23 (1994) 864–894.
9. K. Dabrowski, V. Lozin, R. Raman and B. Ries, Colouring vertices of triangle-free graphs without forests, Discrete Mathematics 312 (2012) 1372–1385.
10. P.A. Golovach and P. Heggernes, Choosability of P_5-free graphs, Proc. MFCS 2009, LNCS 5734 (2009) 82–391.
11. P.A. Golovach, D. Paulusma and B. Ries, Coloring Graphs Characterized by a Forbidden Subgraph, Proc. MFCS 2012, LNCS 7464 (2012) 443–454.
12. P.A. Golovach, D. Paulusma and J. Song, Coloring graphs without short cycles and long induced paths, Proc. FCT 2011, LNCS 6914 (2011) 193–204.
13. P.A. Golovach, D. Paulusma and J. Song, Closing complexity gaps for coloring problems on H-free graphs, Proc. ISAAC 2012, LNCS, to appear.
14. P.A. Golovach, D. Paulusma and J. Song, 4-Coloring H-free graphs when H is small, Discrete Applied Mathematics 161 (2013) 140–150.
15. A. Gyárfás, Problems from the world surrounding perfect graphs, Zastosowania Matematyki Applicationes Mathematicae XIX, 3–4 (1987) 413–441.
16. C.T. Hoàng, F. Maffray and M. Mechebbek, A characterization of b-perfect graphs, Journal of Graph Theory 71 (2012) 95–122.
17. K. Jansen, Complexity Results for the Optimum Cost Chromatic Partition Problem, Universität Trier, Mathematik/Informatik, Forschungsbericht 96–41, 1996.
18. K Jansen and P. Scheffler, Generalized coloring for tree-like graphs, Discrete Appl. Math. 75 (1997) 135–155.
19. D. Kobler and U. Rotics, Edge dominating set and colorings on graphs with fixed clique-width, Discrete Applied Mathematics 126 (2003) 197–221.
20. D. Král', J. Kratochvíl, Zs. Tuza, and G.J. Woeginger, Complexity of coloring graphs without forbidden induced subgraphs, Proc. WG 2001, LNCS 2204 (2001) 254–262.
21. J. Kratochvíl and Z. Tsuza, Algorithmic complexity of list colorings, Discrete Applied Mathematics 50 (1994) 297–302.
22. S. Kratsch and P. Schweitzer, Graph isomorphism for graph classes characterized by two forbidden induced subgraphs, Proc. WG 2012, LNCS 7551 (2012) 34–45.
23. V. V. Lozin, A decidability result for the dominating set problem, Theoretical Computer Science 411 (2010) 4023–4027.
24. F. Maffray and M. Preissmann, On the NP-completeness of the k-colorability problem for triangle-free graphs, Discrete Mathematics 162 (1996) 313–317.
25. G.B. Mertzios and P.G. Spirakis, Algorithms and almost tight results for 3-colorability of small diameter graphs, Proc. SOFSEM 2013, LNCS, to appear.
26. T. R. Jensen and B. Toft, Graph Coloring Problems, Wiley Interscience, 1995.
27. B. Randerath, 3-colorability and forbidden subgraphs. I., Characterizing pairs, Discrete Mathematics 276 (2004) 313–325.
28. B. Randerath and I. Schiermeyer, A note on Brooks' theorem for triangle-free graphs, Australas. J. Combin. 26 (2002) 3–9.
29. B. Randerath and I. Schiermeyer, Vertex colouring and forbidden subgraphs - a survey, Graphs Combin. 20 (2004) 1–40.
30. T. J. Schaefer, The complexity of satisfiability problems, Proc. STOC 1978, 216–226.
31. D. Schindl, Some new hereditary classes where graph coloring remains NP-hard, Discrete Math. 295 (2005)197–202.
32. Zs. Tuza, Graph colorings with local restrictions - a survey, Discuss. Math. Graph Theory 17 (1997) 161–228.