Discrete Applied Mathematics 166 (2014) 123–130

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

List coloring in the absence of two subgraphs[☆]

Petr A. Golovach[a], Daniël Paulusma[b,*]

[a] Department of Informatics, University of Bergen, Norway

[b] School of Engineering and Computing Sciences, Durham University, United Kingdom

ARTICLE INFO

Article history:
Received 8 June 2013
Received in revised form 27 September 2013
Accepted 6 October 2013
Available online 1 November 2013

Keywords:
List coloring
Forbidden induced subgraph
Computational complexity

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 it 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 [27] and Lozin [28] 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 [34] and Tuza [37], and to the book by Jensen and Toft [23].

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 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 $c : V \to \{1, 2, \ldots\}$ *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

☆ This paper was supported by EPSRC (EP/G043434/1) and ERC (267959), and an extended abstract of it appeared in the Proceedings of CIAC 2013.

* Corresponding author. Tel.: +44 0 191 33 41723; fax: +44 0 191 33 41701.
E-mail addresses: petr.golovach@ii.uib.no (P.A. Golovach), daniel.paulusma@durham.ac.uk (D. Paulusma).

0166-218X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dam.2013.10.010

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 let $\{H_1, \ldots, H_p\}$ 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 $K_r - e$ denotes the graph obtained from a complete graph K_r after removing one 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. [25] 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 [15] 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 (for a survey, we refer to the paper of Golovach, Paulusma, and Song [16], and for some new results to a recent paper of Huang [20]). The following theorem gives these three complexity dichotomies.

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 $\ell \le 2$, ℓ-*List Coloring *is polynomial-time solvable. For all $\ell \ge 3$, 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 theorem given below. Here, we let C_3^+ denote the graph with vertices a, b, c, d and edges ab, ac, ad, bc, whereas the graph $\overline{P_1 + P_4}$ is also known as the *gem*. Also note that the graphs H_1 and H_2 may be swapped in each of the subcases of Theorem 2.

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

(i) Coloring *is NP-complete for (H_1, H_2)-free graphs if*
1. *$H_1 \supseteq_i C_r$ for some $r \ge 3$ and $H_2 \supseteq_i C_s$ for some $s \ge 3$,*
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 $r \ge 5$,*
5. *$H_1 \supseteq_i C_r$ for $r \ge 4$ and $H_2 \supseteq_i K_{1,3}$,*
6. *$H_1 \supseteq_i C_3$ and $H_2 \supseteq_i P_{164}$,*
7. *$H_1 \supseteq_i C_r$ for $r \ge 5$ and H_2 contains a spanning subgraph of $2P_2$ as an induced subgraph,*
8. *$H_1 \supseteq_i C_r + P_1$ for $3 \le r \le 4$ or $H_1 \supseteq_i \overline{C_r}$ for $r \ge 6$, and H_2 contains a spanning subgraph of $2P_2$ as an induced subgraph,*
9. *$H_1 \supseteq_i K_4$ or $H_1 \supseteq_i K_4 - e$, and $H_2 \supseteq_i K_{1,3}$.*

(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. *$H_1 \subseteq_i C_3 + P_1$ or $H_1 \subseteq_i 2P_2$, 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 $s \ge 1$,*
5. *$H_1 = K_r$ for $r \ge 4$, and $H_2 \subseteq_i sP_2$ or $H_2 \subseteq_i sP_1 + P_5$ for $s \ge 1$,*
6. *$H_1 \subseteq_i P_1 + P_4$ or $H_1 \subseteq_i P_5$, and $H_2 \subseteq_i \overline{P_1 + P_4}$,*
7. *$H_1 \subseteq_i P_1 + P_4$ or $H_1 \subseteq_i 2P_2$, and $H_2 \subseteq_i \overline{P_5}$,*
8. *$H_1 \subseteq_i K_r - e$ for $r \ge 2$, and $H_2 \subseteq_i sP_1 + P_2$ for $s \ge 0$ or $H_2 \subseteq_i 2P_2$.*