Three complexity results on coloring P_k-free graphs

Hajo Broersma[1], Fedor V. Fomin[2], Petr A. Golovach[2], and Daniël Paulusma[1]

[1] Department of Computer Science, Durham University, DH1 3LE Durham, United Kingdom,
{hajo.broersma,daniel.paulusma}@durham.ac.uk ⋆

[2] Department of Informatics, University of Bergen, PB 7803, 5020 Bergen, Norway,
{fedor.fomin,petr.golovach}@ii.uib.no ⋆⋆

Abstract. We prove three complexity results on vertex coloring problems restricted to P_k-free graphs, i.e., graphs that do not contain a path on k vertices as an induced subgraph. First of all, we show that the pre-coloring extension version of 5-coloring remains NP-complete when restricted to P_6-free graphs. Recent results of Hoàng et al. imply that this problem is polynomially solvable on P_5-free graphs. Secondly, we show that the pre-coloring extension version of 3-coloring is polynomially solvable for P_6-free graphs. This implies a simpler algorithm for checking the 3-colorability of P_6-free graphs than the algorithm given by Randerath and Schiermeyer. Finally, we prove that 6-coloring is NP-complete for P_7-free graphs. This problem was known to be polynomially solvable for P_5-free graphs and NP-complete for P_8-free graphs, so there remains one open case.

1 Introduction

In this paper we consider computational complexity issues related to vertex coloring problems restricted to P_k-free graphs. Due to the fact that the usual vertex ℓ-coloring problem is NP-complete for any fixed $\ell \geq 3$, there has been considerable interest in studying its complexity when restricted to certain graph classes. Without doubt one of the most well-known results in this respect is that ℓ-coloring is polynomially solvable for perfect graphs. More information on this classic result and related work on coloring problems restricted to graph classes can be found in, e.g., [11] and [13]. Instead of repeating what has been written in so many papers over the years, we also refer to these surveys for motivation and background. Here we continue the study of ℓ-coloring and its variants for P_k-free graphs, a problem that has been studied in several earlier papers by different groups of researchers (see, e.g., [4], [8], [9], [10], [14]).

2 Background and terminology

We refer to [1] for standard graph theory terminology and to [3] for terminology on computational complexity. Let $G = (V, E)$ be a graph. A (vertex) coloring of G is a mapping $\phi : V \to \{1, 2, \ldots\}$ such that $\phi(u) \neq \phi(v)$ whenever $uv \in E$. Here $\phi(u)$ is usually referred to as the color of u in the coloring ϕ of G. An ℓ-coloring of G

⋆ The work of DP is supported by EPSRC Grant EP/D053633/1.

⋆⋆ Supported by the Norwegian Research Council.

is a mapping $\phi : V \to \{1, 2, \ldots, \ell\}$ such that $\phi(u) \neq \phi(v)$ whenever $uv \in E$. In list-coloring we assume that $V = \{v_1, v_2, \ldots, v_n\}$ and that for every vertex v_i of G there is a list L_i of admissible colors (a subset of the natural numbers). Given these lists, a list-coloring of G is a mapping $\phi : V \to \{1, 2, \ldots\}$ such that $\phi(v_i) \in L_i$ for all $i \in \{1, 2, \ldots, n\}$ and $\phi(v_i) \neq \phi(v_j)$ whenever $v_iv_j \in E$. In pre-coloring extension we assume that a (possible empty) subset $W \subseteq V$ of G is pre-colored with $\phi_W : W \to \{1, 2, \ldots\}$ and the question is whether we can extend ϕ_W to a coloring of G. If ϕ_W is restricted to $\{1, 2, \ldots, \ell\}$ and we want to extend it to an ℓ-coloring of G, we say we deal with the pre-coloring extension version of ℓ-coloring. In fact, we consider a slight variation on the latter problem which can be considered as list coloring, but which has the flavor of pre-coloring: lists have varying sizes including some of size 1. We will slightly abuse terminology and call these problems pre-coloring extension problems too.

3 Results of this paper

We prove the following three complexity results on vertex coloring problems restricted to P_k-free graphs.

- First of all, in Section 4 we show that the pre-coloring extension version of 5-coloring remains NP-complete when restricted to P_6-free graphs. Recent results of Hoàng et al. [4] imply that this problem is polynomially solvable on P_5-free graphs. Their algorithm for ℓ-coloring for any fixed ℓ is in fact a list-coloring algorithm where the lists are from the set $\{1, 2, \ldots, \ell\}$.
- Secondly, in Section 5 we show that the pre-coloring extension version of 3-coloring is polynomially solvable for P_6-free graphs. The 3-coloring problem was known to be polynomially solvable for P_6-free graphs from [10], where the authors use the Strong Perfect Graph Theorem and a result of Tucker [12] to obtain their algorithm. Our algorithm is simpler, independent of the Strong Perfect Graph Theorem, and uses a recent structural result of [5]. We reduce the 8 page journal description of the algorithm in [10] to 3 pages.
- Finally, in Section 6 we prove that 6-coloring is NP-complete for P_7-free graphs. This problem was known to be polynomially solvable for P_5-free graphs [4] and NP-complete for P_8-free graphs [14], so there remains one open case.

4 Pre-coloring extension of 5-coloring for P_6-free graphs

In this section we show that the pre-coloring extension version of 5-coloring remains NP-complete when restricted to P_6-free graphs. We use a reduction from not-all-equal 3-Satisfiability (NAE 3SAT) with positive literals only (also known as HYPERGRAPH 2-COLORABILITY), which is defined as follows. Given a set $X = \{x_1, x_2, \ldots, x_n\}$ of logical variables, and a set $C = \{C_1, C_2, \ldots, C_m\}$ of three-literal 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?

We consider an arbitrary instance I of NAE 3SAT and define a graph G_I and a pre-coloring on some vertices of G_I, and next we show that G_I is P_6-free and that the pre-coloring on G_I can be extended to a 5-coloring of G_I if and only if I has a satisfying truth assignment in which each clause contains at least one true literal and at least one false literal.

4.1 The graph G_I corresponding to the instance I

Let I be an arbitrary instance of NAE 3SAT with variables $\{x_1, x_2, \dots, x_n\}$ and clauses $\{C_1, C_2, \dots, C_m\}$. We define a graph G_I corresponding to I and lists of admissible colors for its vertices based on the following construction. We note here that the lists we introduce below are only there for convenience to the reader; it will be clear later that all lists other than $\{1, 2, \dots, 5\}$ are in fact forced by the pre-colored vertices.

- We introduce one new vertex for each of the clauses, and use the same labels $C_1, C_2, \dots, C_m$ for these m vertices; we assume that for each of these vertices there is a list $\{1, 2, 3\}$ of admissible colors. We say that these vertices are of C-type.
- We introduce one new vertex for each of the variables, and use the same labels $x_1, x_2, \dots, x_n$ for these n vertices; we assume that for each of these vertices there is a list $\{4, 5\}$ of admissible colors. We say that these vertices are of x-type.
- We join all C-type vertices to all x-type vertices to form a large complete bipartite graph.
- For each clause C_j containing the variables x_i, x_k and x_r we introduce three pairs of new vertices $\{a_{i,j}, b_{i,j}\}$, $\{a_{k,j}, b_{k,j}\}$, $\{a_{r,j}, b_{r,j}\}$; we assume the following lists of admissible colors for these three pairs, respectively: $\{\{1, 4\}, \{2, 5\}\}$, $\{\{2, 4\}, \{3, 5\}\}$, $\{\{3, 4\}, \{1, 5\}\}$. We say that these vertices are of a-type and b-type. We add edges between x-type and a-type vertices whenever the first index of the a-type vertex is the same as of the x-type vertex, and similarly for the b-type vertices. We add edges between C-type and a-type vertices whenever the second index of the a-type vertex is the same as the index of the C-type vertex, and similarly for the b-type vertices. Hence each clause with three variables is represented by three 4-cycles that have one C-type vertex in common.
- For each a-type vertex we introduce a new copy of a $K_{2,3}$, as follows: for $a_{i,j}$ we introduce five new vertices $\{p_{i,j,1}, \dots, p_{i,j,5}\}$, and we add all edges between $\{p_{i,j,1}, p_{i,j,2}, p_{i,j,3}\}$ and $\{p_{i,j,4}, p_{i,j,5}\}$. We say that these vertices are of p-type. We add edges between each a-vertex and the p-vertices of its corresponding $K_{2,3}$ depending on its list of admissible colors. In particular, we join the a-vertex to the three p-vertices of its $K_{2,3}$ that have a third index which is not in its list of admissible colors. So, if $a_{i,j}$ has list $\{1, 4\}$, we join it to $p_{i,j,2}, p_{i,j,3}, p_{i,j,5}$.
- For each b-type vertex we introduce a new copy of a $K_{2,3}$ on five vertices of q-type, in the same way as we introduced the p-type vertices for the a-type vertices. Edges are added in a similar way, depending on the indices and the lists.
- We join all the p-type and q-type vertices with third indices $1, 2, 3$ to all the p-type and q-type vertices with third indices $4, 5$ to form a huge complete bipartite graph.
- We join all x-type vertices to all p-type and q-type vertices with third indices $1, 2, 3$.

- We join all C-type vertices to all p-type and q-type vertices with third indices $4, 5$.
- We pre-color all the p-type and q-type vertices according to their third index, so $p_{i,j,\ell}$ will be pre-colored with color $\ell \in \{1, 2, \ldots, 5\}$. Note that we can now in fact replace all lists introduced earlier by $\{1, 2, \ldots, 5\}$, since the shorter lists will be forced by the given pre-coloring.

4.2 The proofs for the result on 5-coloring

Lemma 1. *The graph G_I is P_6-free.*

Proof. We give a proof by contradiction. Suppose the graph G_I contains an induced subgraph H which is isomorphic to a P_6. Then it is obvious that H contains at most three vertices from the set S of all p-type and q-type vertices; otherwise H would either contain a cycle, or an independent set of four vertices, or a vertex with degree at least three. Analogously, H contains at most three vertices from the set T of all C-type and x-type vertices. By similar arguments, it is not difficult to show that H contains at most three vertices from $S \cup T$. We complete the proof by a series of claims followed by proofs.

Claim 1. H contains at most two vertices of S.

Proof of Claim 1. Suppose $|V(H) \cap S| = 3$. This implies H does not contain a vertex of T, so H contains three vertices from the set U of all a-type and b-type vertices. This is obviously impossible and completes the proof of Claim 1.

Claim 2. H contains at most one vertex of S.

Proof of Claim 2. Suppose $|V(H) \cap S| = 2$. Then clearly H contains at least one vertex of T; otherwise $|V(H)| \leq 4$. So $|V(H) \cap T| = 1$, and H contains three vertices of U. If $V(H) \cap S$ is an adjacent pair, the vertex of $V(H) \cap T$ is adjacent to precisely one of them, and we easily obtain a contradiction. In the other case, $(V(H) \cap S) \cup (V(H) \cap T)$ induces either a P_3 or an independent set in H. Both cases lead to contradictions, as is easily seen. This completes the proof of Claim 2.

Claim 3. H contains no vertex of S.

Proof of Claim 3. Suppose $|V(H) \cap S| = 1$. Then clearly, $|V(H) \cap T| = 1$ or $|V(H) \cap T| = 2$. The first case is impossible since U is an independent set. For the second case first observe that common neighbors of two vertices from U can only be in T, and that two x-type vertices or two C-type vertices do not have a common neighbor in U. Noting that the three vertices of $V(H) \cap U$ form an independent set, there are two possibilities for the remaining three vertices of H: they either induce an independent set in H or a P_2 and a P_1. In the first case the two vertices of $V(H) \cap T$ are either both x-type or both C-type vertices. This easily yields a contradiction. In the second case, the induced P_1 can only result from a vertex in T, so the P_2 is induced by a vertex from S and a vertex from T. Now clearly the two vertices from $V(H) \cap T$ must be both of x-type or both of C-type. But then both these vertices are adjacent to the vertex of $V(H) \cap S$, a contradiction. This completes the proof of Claim 3.

We conclude that H contains no vertex of S and at most three vertices of T. So H contains at least three vertices of U which form an independent set in H. This yields only one case: H contains precisely three vertices of T and precisely three vertices of U. By previous observations all vertices of $V(H) \cap T$ must be of the same type, so they form an independent set as well. Recalling that two x-type vertices or two C-type vertices have no common neighbors in U, we obtain a contradiction. This completes the proof of Lemma 1. $\square$

Lemma 2. *If I has a satisfying truth assignment in which each clause contains at least one true and at least one false literal, then the pre-coloring of G_I can be extended to a 5-coloring of G_I.*

Proof. Suppose I has a satisfying truth assignment in which each clause contains at least one true and at least one false literal. Assume we have the pre-coloring on the p-type and q-type vertices, and the lists of admissible colors as indicated in the previous section. We use color 4 to color the x-type vertices representing the true literals and color 5 for the false literals. Now consider the lists assigned to the a-type and b-type vertices that come in pairs chosen from $\{\{1,4\},\{2,5\}\}$, $\{\{2,4\},\{3,5\}\}$, $\{\{3,4\},\{1,5\}\}$. If the adjacent x-type vertex has color 4, color 1, 2 or 3 is forced on one of the adjacent a-type or b-type vertices, respectively, while on the other one we can use color 5; similarly, if the adjacent x-type vertex has color 5, color 2, 3 or 1 is forced on one of the adjacent a-type or b-type vertices, respectively, while on the other one we can use color 4. Since precisely two of the three x-type vertices of one clause gadget have the same color, this leaves at least one of the colors 1, 2 and 3 admissible for the C-type vertex representing the clause. It is clear that we can obtain a 5-coloring for the pre-colored graph G_I. This completes the proof of Lemma 2. $\square$

Lemma 3. *If the pre-coloring of G_I can be extended to a 5-coloring of G_I, then I has a satisfying truth assignment in which each clause contains at least one true and at least one false literal.*

Proof. Suppose we have a 5-coloring of the graph G_I that respects the pre-coloring and all lists assigned as indicated in the previous section. Then each of the x-type vertices has color 4 or 5, and each of the C-type vertices has color 1, 2 or 3. We define a truth assignment that sets a variable to TRUE if the corresponding x-type vertex has color 4, and to FALSE otherwise. Suppose one of the clauses contains only true literals. Then the three x-type vertices in the corresponding clause gadget of G_I all have color 4. Now consider the lists assigned to the a-type and b-type vertices of this gadget that come in pairs chosen from $\{\{1,4\},\{2,5\}\}$, $\{\{2,4\},\{3,5\}\}$, $\{\{3,4\},\{1,5\}\}$. Since the adjacent x-type vertices all have color 4, colors 1, 2 and 3 are forced on three of the a-type and b-type vertices adjacent to the C-type vertex of this gadget, a contradiction, since the C-type vertex has color 1, 2 or 3. This proves that every clause contains at least one false literal. Analogously, it is easy to show that every clause contains at least one true literal. This completes the proof of Lemma 3. $\square$

5 Pre-coloring extension of 3-coloring for P_6-free graphs

In this section we show that the pre-coloring extension version of 3-coloring is polynomially solvable for P_6-free graphs. The key ingredient in our approach is the following recently obtained characterization of P_6-free graphs [5]. Here a subgraph H of a graph G is said to be a dominating subgraph of G if every vertex of $V(G) \setminus V(H)$ has a neighbor in H.

Theorem 1 ([5]). *A graph G is P_6-free if and only if each connected induced subgraph of G on more than one vertex contains a dominating induced cycle on six vertices or a dominating (not necessarily induced) complete bipartite subgraph. Moreover, these dominating subgraphs can be obtained in polynomial time.*

A key ingredient in our approach is the following observation: it is checkable in polynomial time whether a pre-coloring of a graph G can be extended to a proper ℓ-coloring of G as soon as the uncolored vertices of G have admissible lists of size at most 2. In this case the remaining decision problem can be modeled and solved as a $2\mathrm{SAT}$-problem. This approach has been introduced by Edwards [2] and is folklore now. It has been used especially for checking 3-colorability of graphs with small dominating sets (if such dominating sets can be found in polynomial time), e.g., for P_5-free graphs ([4]) and for P_6-free graphs ([10]). By exhaustively checking all possible 3-colorings on the dominating set combined with solving the $2\mathrm{SAT}$-problem(s) on the remaining (dominated) vertices, this yields a polynomial time algorithm.

This obviously solves our problem in case the (component of the) instance graph contains a dominating C_6: all lists of admissible colors on the vertices in the beginning are subsets of $\{1, 2, 3\}$ and after assuming a coloring on the C_6 (respecting the pre-coloring, i.e., lists of size 1) all lists of admissible colors for the uncolored vertices have size at most 2, and we can model and solve the remaining problem as a $2\mathrm{SAT}$-problem. Although in the other case we cannot assume that the dominating complete bipartite graph has a bounded size, we can use a similar approach due to the special structure of P_6-free graphs. We will describe the procedure in more detail.

Suppose our instance graph G is connected (otherwise we treat the components of G separately), that we have lists of admissible colors from the set $\{1, 2, 3\}$ on each vertex of G, and that we have constructed a dominating complete (not necessarily induced) bipartite graph H of G with bipartition classes A and B.

If there exists no 3-coloring of G (respecting a possible pre-coloring, i.e., respecting the given lists) in which one of A and B is monochromatic (i.e., every vertex of A or B receives the same color), then clearly G has no 3-coloring extending the pre-coloring, since we have to use at least 4 colors on H. Then we eventually obtain a NO answer after first trying all cases with A monochromatic and successively with B monochromatic, in the way we describe below.

Hence we can assume that A or B is monochromatic, and we can guess that A is monochromatic (if this does not result in a 3-coloring of G we can repeat the procedure assuming that B is monochromatic).

From now on we assume that all vertices of A are colored with color 1 (possibly after renaming the colors). We remove color 1 from all the lists of admissible colors at vertices of $N(A) = \bigcup_{v \in A} N(v) \setminus A$, we choose one vertex $a \in A$ and delete all vertices of $A \setminus \{a\}$. We let R denote the subset of all remaining vertices with admissible lists of size 3. Clearly we are done with the graph (or component) G if $R = \emptyset$, as argued above, simply by solving a 2SAT-problem defined on the uncolored vertices and all edges incident with these vertices.

So let us assume $R \neq \emptyset$. Clearly $B \cap R = \emptyset$ because all vertices in B have a neighbor colored with color 1, so their admissible lists have size at most 2. It is now also clear that B dominates R (since A does not dominate any vertex of R; otherwise the list of such a vertex would have been updated to size at most 2). Now let us consider the subgraph Q of $G' = G - (A \setminus \{a\})$ induced by the vertices of $V(G') \setminus (\{a\} \cup B)$. If Q contains an isolated vertex v (i.e., a vertex with no neighbors in Q) such that $v \in R$, then we can use color 1 on v and remove v. So we can assume that all isolated vertices of Q have admissible lists of size at most 2. We next analyze pairs of adjacent vertices of Q, and distinguish a number of cases.

Case 1. Q contains an edge pq such that p is adjacent to a vertex $b \in B \setminus N(q)$ and q is adjacent to a vertex $c \in B \setminus N(p)$.

First note that the set $S = \{a, b, c, p, q\}$ induces a C_5 with possibly an additional edge bc in G'. If S dominates all vertices of R, we can just guess the eligible 3-colorings on S and solve our problem for the graph G' by solving a polynomial number of 2SAT-problems.

Supposing the contrary, let $x \in R$ be a vertex that is not dominated by S. Since B dominates R there exists a vertex $y \in B \setminus S$ with $xy \in E(G')$. Consider the paths $xyabpq$ and $xyacqp$ on six vertices. If $yb \in E(G')$ or $yc \in E(G')$, then guessing a 3-coloring on S would also fix the eligible color on y, and reduce the list size on x. So if this would occur for all possible choices of x and y, we could solve our problem in polynomial time. It remains to consider the cases where $yb \notin E(G')$ and $yc \notin E(G')$. Now since G' is P_6-free at least one of $\{yp, yq\}$ is an edge of G'. If both are edges of G', then, since in any 3-coloring of G' at least one of p and q receives color 2 or 3, any eligible 3-coloring on S will fix the eligible color on y, and reduce the list size on x. An analogous situation occurs when x, a and p share a common neighbor, and x, a and q share another common neighbor.

We next analyze the subcase that there are pairs of vertices $p', q' \in R$ not dominated by S, but where p', a, p have a common neighbor b' and q', a, q have a common neighbor $c' \neq b'$ such that b' is not adjacent to either of $\{b, c, q, q'\}$ and c' is not adjacent to either of $\{b, c, p, p'\}$. Now consider the path $p'b'pqc'q'$. If $b'c' \in E(G')$, then in any 3-coloring on S at least one of p and q receives color 2 or 3, and the eligible colors on b' and c' will be fixed, and the lists on p' and q' reduced. So we can deal comfortably with this subcase. Since G' is P_6-free, assuming $b'c' \notin E(G')$ the only other possible subcase is that $p'q' \in E(G')$. But this yields a contradiction, since then $\{q, c, a, b', p', q'\}$ induces a P_6 in G'.

For the remainder of Case 1, we can now assume that the only subcase that has to be resolved is when all vertices of R that are not dominated by S (like x above) have no neighbor in common with both a and p, but only with a and q, or symmetrically. Then we can use the same approach as before if q receives color 2 or 3 in the guessed 3-coloring on S. If this does not result in a 3-coloring of G' in the end, we start the whole procedure (with color 1 on each vertex of A) again after assigning color 1 to q, adjusting the lists on all vertices in $N(q)$, and removing the vertex q. This clearly does not result in a combinatorial explosion in the sense of the computational complexity.

Concluding, for all subcases we analyzed in Case 1 except for one, we can propagate any 3-coloring on the set S to obtain a reduction of the list sizes of all vertices in R, and solve our problem using 2SAT-formulations (or obtain an obstruction to a 3-coloring at an earlier stage). In the other subcase, we either also get such a reduction or we can pre-color a specific new vertex and start the procedure on a smaller instance.

In the next case we assume that Case 1 does not apply.

Case 2. Q contains an edge pq such that p is adjacent to a vertex $b \in B \cap N(q)$ and q is adjacent to a vertex $c \in B \setminus N(p)$.

The set $S = \{a, b, c, p, q\}$ now induces a C_5 with an edge bq and possibly an additional edge bc in G'. If S dominates all vertices of R, we are done as before. So let $x \in R$ and $y \in N(x) \cap B$ be defined as before. Just like in Case 1, if $cy \in E(G')$ a 3-coloring on S will fix the eligible color on y and reduce the list of x. The arguments of Case 1 can also be copied if both yp and yq are edges in G'. If $yp \in E(G')$ and $yq \notin E(G')$, we contradict our assumptions since we are in Case 1 with y instead of b. Considering the subgraph induced by $\{x, y, a, c, q, p\}$ and using that G' is P_6-free we conclude that in the remaining subcase $yq \in E(G')$ and $yp \notin E(G')$. Analogously to the exceptional subcase of Case 1, we can use the same approach as before if q receives color 2 or 3 in the guessed 3-coloring on S. If this does not result in a 3-coloring of G' in the end, we start the whole procedure (with color 1 on each vertex of A) again after assigning color 1 to q, adjusting the lists on all vertices in $N(q)$, and removing the vertex q.

We arrive at the same conclusion as in Case 1.

In the remainder we assume that neither Case 1 nor Case 2 applies. This implies that for each edge pq in Q, the vertices p and q have exactly the same neighbors in B, so by repeating the arguments this holds for all vertices in the same component of Q.

Case 3. All vertices in each component of Q have the same neighbors in B.

We start with the graph G' as above. As long as there exist or appear new vertices with lists of size 1 that are not in $B \cup \{a\}$, we do the following: for such a vertex v we adjust the lists of all vertices of $N(v)$, and then remove v (unless we can conclude that we cannot obtain a 3-coloring of G' extending the pre-coloring; then we stop and return to an earlier stage with a different guess on S or finally on B). Denote the resulting graph by G^*, and assume that in the remainder all neighborhoods, lists of admissible colors, subsets of vertices, etc. are with respect to G^*. In particular, let Q be the subgraph of G^* induced by the vertices of $V(G^*) \setminus (\{a\} \cup B)$. Recall that if Q contains an isolated vertex v with a list containing color 1, then we can use color 1 on v and remove v. So we can assume that all isolated vertices of Q have admissible lists not containing color 1.

Consider the set $B' \subseteq B$ with vertices that have lists $\{2, 3\}$; the other vertices of B have a fixed color, so every vertex dominated by such a vertex has a list of size 2.

Suppose C is a component of the subgraph $G^*[B']$ induced by B' in G^*. Then clearly C is a bipartite graph (otherwise G^* is trivially not 3-colorable and we can recognize this) with all lists equal to $\{2, 3\}$. So if we fix one color on a vertex of C, the other colors on C will also be fixed. If C' is another component of $G^*[B']$ such that C and C' are connected by a path with internal vertices in Q, then fixing one color on a vertex of C will also fix the colors on C': this is clear if C and C' have an isolated vertex v of Q as a common neighbor, since the list of v does not contain color 1; in the other case, it follows from the assumption that all vertices in each component of Q have the same neighbors in B, so the colors propagate from C to C' through subgraphs isomorphic to K_4 minus an edge. We can split the checking whether the pre-coloring can be extended to a 3-coloring of G^* in separate disjoint problems now. Let $\mathcal{C}$ denote a maximal set of components of $G^*[B']$ that are connected by paths with internal vertices in Q that force the propagation of one fixed color in $\mathcal{C}$ to fixed colors for all vertices in $\mathcal{C}$. Let $\mathcal{D}$ denote all vertices of Q dominated by vertices of $\mathcal{C}$. Then fixing one color (so all colors) on $\mathcal{C}$, we can model the problem on $\mathcal{D}$ as a 2SAT-problem. If this results in a YES answer, we can check the next maximal set of components, etc.; if for one of the sets we get a NO answer, we try the swap of colors on this set; if we still get a NO answer, we repeat the whole procedure with color 1 on all vertices of B instead of A.

One readily checks that the above arguments can be turned into a polynomial algorithm for checking whether a pre-coloring on a P_6-free graph G can be extended to a 3-coloring of G.

6 6-Coloring for P_7-free graphs

In this section we prove that 6-coloring is NP-complete for P_7-free graphs. We use a reduction from 3-Satisfiability (3SAT).

We consider an arbitrary instance I of 3SAT and define a graph G_I, and next we show that G_I is P_7-free and that G_I is 6-colorable if and only if I has a satisfying truth assignment.

6.1 The graph G_I corresponding to the instance I

Let I be an arbitrary instance of 3SAT with variables $\{x_1, x_2, \ldots, x_n\}$ and clauses $\{C_1, C_2, \ldots, C_m\}$. We define a graph G_I corresponding to I based on the following construction.

- We introduce a gadget on 8 new vertices for each of the clauses, as follows: for clause C_j we introduce a gadget with vertex set:
  $\{a_{j,1}, a_{j,2}, a_{j,3}, b_{j,1}, b_{j,2}, b_{j,3}, c_{j,1}, c_{j,2}\}$ and edge set:
  $\{a_{j,1}a_{j,2}, a_{j,1}a_{j,3}, a_{j,2}a_{j,3}, a_{j,1}b_{j,1}, a_{j,2}b_{j,2}, a_{j,3}b_{j,3}, b_{j,1}c_{j,1}, b_{j,1}c_{j,2}, b_{j,2}c_{j,1}, b_{j,2}c_{j,2}, b_{j,3}c_{j,1}, b_{j,3}c_{j,2}, c_{j,1}c_{j,2}\}$.
  We say that these vertices are of a-type, b-type and c-type.
- We introduce a gadget on 3 new vertices for each of the variables, as follows: for variable x_i we introduce a complete graph with vertex set $\{x_i, \overline{x}_i, y_i\}$. We say that these vertices are of x-type (both the x_i and the $\overline{x}_i$ vertices) and of y-type.
- If clause C_j contains the variables x_i, x_k and x_r, we add three matching edges between the corresponding literal vertices (so x_i or $\overline{x}_i$, etc., depending on which of them appear in C_j) and the three b-type vertices of the gadget corresponding to

C_j. If $b_{j,s}x_i$ or $b_{j,s}\overline{x}_i$ has been added as an edge, we also add the edge $b_{j,s}y_i$, and analogously for x_k and x_r.

- We introduce three additional vertices d_1, d_2 and z, and join d_1 and d_2 by an edge. We join all x_i to d_1 by edges, and all $\overline{x}_i$ to d_2.
- We join z to all vertices of y-type, a-type and c-type, and to d_1 and d_2.
- We join all the x-type vertices and y-type vertices to all the a-type and c-type vertices.
- Finally, we join d_1 and d_2 to all the a-type, b-type and c-type vertices.

6.2 The proofs for our result on 6-coloring

Lemma 4. *The graph G_I is P_7-free.*

Proof. We give a proof by contradiction. Suppose the graph G_I contains an induced subgraph H which is isomorphic to a P_7. We complete the proof by a series of claims followed by proofs. We only give a sketch of the proofs here due to page restrictions.

Claim 1. H does not contain vertex z.

Proof of Claim 1. Suppose H contains z. Then, since there are no cycles in H, we know that an a-type or a c-type vertex can not be in H together with d_1 or d_2, or with an y-type vertex. Using this one readily shows that H can neither contain d_1 nor d_2, since the edge zd_i cannot be extended to a P_7. Using this information, by analogous arguments one can show that H contains no y-type vertices. Next one can analyze the case that z is adjacent to an a-type vertex in H, and finally that z is adjacent to a c-type vertex in H, to obtain contradictions. This completes the proof of Claim 1.

Claim 2. H contains at most one of the vertices d_1 and d_2.

Proof of Claim 2. Suppose H contains both d_1 and d_2. Then H does not contain any vertices of a-type, b-type or c-type, and at most two x-type vertices (with one positive and one negative literal in the case it contains two). The longest path we can obtain is a P_6. This completes the proof of Claim 2.

Claim 3. H contains at most one of the y-type vertices.

Proof of Claim 3. Suppose H contains at least two of the y-type vertices. Then clearly H contains at most one from all the a-type and c-type vertices. From the y-vertices we can only extend paths with x-type or b-type vertices. One readily shows that we cannot obtain a P_7. This completes the proof of Claim 3.

In a similar fashion one can show successively that H contains at most one a-type vertex, at most one c-type vertex, at most one x-type vertex, and that an a-type and a c-type vertex do not appear together in H. Using all the above information, we then conclude that H contains at least three b-type vertices. This implies that neither d_1 nor d_2 are contained in H, so H contains at least four b-type vertices. It is now easy to obtain a contradiction. This completes the proof of Lemma 4. □

Lemma 5. *If I has a satisfying truth assignment, then G_I is 6-colorable.*

Proof. Suppose I has a satisfying truth assignment. We use color 4 or 5 to color the x-type vertices representing the true literals and color 6 for the false literals. In particular, if x_i is true, we use color 5 to color the corresponding vertex; if $\overline{x}_i$ is true, we use color 4 to color the corresponding vertex. We use color 4 or color 5 to color the y-type vertices, depending on the colors we used for the x-type vertices. This clearly yields a proper 3-coloring of all the variable gadgets with colors 4, 5 and 6. We extend this 3-coloring by using color 6 for z and colors 4 and 5 for d_1 and d_2, respectively. For the true literals of C_j, we can use color 6 for the corresponding b-type vertex, and color 1 for the other b-type vertices of the corresponding clause gadget. Since each clause contains at least one true literal, we note that we do not use color 1 for all three b-type vertices of the clause gadgets. We can now use colors 2 and 3 for the c-type vertices and colors 1, 2 and 3 for the a-type vertices to extend the coloring to a 6-coloring of G_I. This completes the proof of Lemma 5. □

Lemma 6. *If G_I is 6-colorable, then I has a satisfying truth assignment.*

Proof. Suppose we have a 6-coloring of the graph G_I with colors $\{1, 2, \ldots 6\}$. We assume that vertex z has color 6, that d_2 has color 5, and that d_1 has color 4. This implies all a-type and c-type vertices have colors from $\{1, 2, 3\}$, and all three colors are used on the a-type vertices, and two of the three on the c-type vertices. This implies that all x-type vertices have colors from $\{4, 5, 6\}$ and all y-type vertices from $\{4, 5\}$. Without loss of generality, suppose that in one of the clause gadgets, the c-type vertices have color 2 and 3. Then the b-type vertices in this clause gadget can only have colors from $\{1, 6\}$. If all of them have color 1, we obtain a contradiction with the coloring of the three a-type vertices in this gadget. So at least one of the b-type vertices has color 6. The same holds if we would have assumed another choice for the two colors used on the c-type vertices. This implies that the corresponding x-type vertex has color 4 or 5. We define a truth assignment that sets a literal to FALSE if the corresponding x-type vertex has color 6, and to TRUE otherwise. It is clear that we obtain a satisfying truth assignment for I. This completes the proof of Lemma 6. □

7 Conclusions and open problems

We proved that the pre-coloring extension version of 5-coloring remains NP-complete when restricted to P_6-free graphs. Recent results of Hoàng et al. [4] imply that this problem is polynomially solvable on P_5-free graphs. In fact they show that ℓ-coloring for any fixed ℓ is polynomially solvable on P_5-free graphs. We note here that in contrast determining the chromatic number (i.e., the smallest ℓ such that the graph is ℓ-colorable) is NP-hard on P_5-free graphs [7]. We also showed that the pre-coloring extension version of 3-coloring is polynomially solvable for P_6-free graphs. Finally, we proved that 6-coloring is NP-complete for P_7-free graphs. This problem was known to be polynomially solvable for P_5-free graphs and NP-complete for P_8-free graphs. This leaves the natural open problem for 6-coloring on P_6-free graphs. Also the complexity of 4-coloring and 5-coloring on P_6-free graphs are open problems. We refer to [9] for the most recent table of the complexity status of ℓ-coloring for P_k-free graphs: the problem is trivially in P for arbitrary fixed ℓ if $k \leq 2$; it is also in P for fixed $k \leq 5$ and arbitrary fixed ℓ, and for $k = 6$ and $\ell = 3$; it is NP-complete for $\ell = 4$ and any $k \geq 9$, for $\ell = 5$ and $k \geq 8$, for $\ell \geq 6$ and $k \geq 8$ (and by our result also for $k = 7$). Interesting questions are: what is the complexity of 4-coloring for P_6-free graphs, of 3-coloring for P_7-free graphs; does there exist an integer k such that 3-coloring is NP-complete for P_k-free graphs? What is the complexity of 5-coloring for P_7-free graphs, and of 4-coloring for P_8-free graphs? We finish this paper with two other open problems on 3-coloring that have intrigued many researchers: the complexity of 3-coloring is open for graphs with diameter 2, and for graphs with diameter 3.

References

1. J.A. Bondy and U.S.R. Murty, Graph Theory, Springer Graduate Texts in Mathematics 244 (2008).
2. K. Edwards, The complexity of coloring problems on dense graphs. Theor. Comp. Science 43, 337–343 (1986).
3. M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, San Francisco (1979).
4. C.T. Hoàng, M. Kamiński, V. Lozin, J. Sawada, and X. Shu, Deciding k-colorability of P_5-free graphs in polynomial time, Algorithmica, in press (DOI link: 10.1007/s00453-008-9197-8).
5. P. van 't Hof and D. Paulusma, A new characterization of P_6-free graphs, Discrete Appl. Math., in press (DOI link: 10.1016/j.dam.2008.08.025).
6. M. Hujter and Zs. Tuza, Precoloring extension II: graph classes related to bipartite graphs, Acta Math. Univ. Comen. 62, 1–11 (1993).
7. D. Král, J. Kratochvíl, Zs. Tuza, and G.J. Woeginger, Complexity of coloring graphs without forbidden induced subgraphs, Lecture Notes in Computer Science 2204, 254–262 (2001).
8. J. Kratochvíl, Precoloring extension with fixed color bound, Acta Math. Univ. Comen. 62, 139–153 (1993).
9. V.B. Le, B. Randerath and I. Schiermeyer, On the complexity of 4-coloring graphs without long induced paths, Theor. Comp. Science 389, 330–335 (2007).
10. B. Randerath and I. Schiermeyer, 3-Colorability $\in P$ for P_6-free graphs, Discrete Appl. Math. 136, 299-313 (2004).
11. B. Randerath and I. Schiermeyer, Vertex colouring and forbidden subgraphs - a survey, Graphs and Combin. 20, 1–40 (2004).
12. A. Tucker, A reduction procedure for coloring perfect K_4-free graphs, J. Combin. Theory B 43, 151–172 (1987).
13. Zs. Tuza, Graph colorings with local restrictions - a survey, Discuss. Math. Graph Theory 17, 161–228 (1997).
14. G.J. Woeginger and J. Sgall, The complexity of coloring graphs without long induced paths, Acta Cybern. 15 , 107-117 (2001).