Minimal Obstructions to C_5-Coloring in Hereditary Graph Classes

Jan Goedgebeur
Department of Computer Science, KU Leuven, Kortrijk, Belgium
Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Belgium

Jorik Jooken
Department of Computer Science, KU Leuven, Kortrijk, Belgium

Karolina Okrasa
Warsaw University of Technology, Poland

Paweł Rzążewski
Warsaw University of Technology, Poland
University of Warsaw, Poland

Oliver Schaudt
University of Cologne, Germany

Abstract

For graphs G and H, an H-coloring of G is an edge-preserving mapping from $V(G)$ to $V(H)$. Note that if H is the triangle, then H-colorings are equivalent to 3-colorings. In this paper we are interested in the case that H is the five-vertex cycle C_5.

A minimal obstruction to C_5-coloring is a graph that does not have a C_5-coloring, but every proper induced subgraph thereof has a C_5-coloring. In this paper we are interested in minimal obstructions to C_5-coloring in F-free graphs, i.e., graphs that exclude some fixed graph F as an induced subgraph. Let P_t denote the path on t vertices, and let $S_{a,b,c}$ denote the graph obtained from paths $P_{a+1}, P_{b+1}, P_{c+1}$ by identifying one of their endvertices.

We show that there is only a finite number of minimal obstructions to C_5-coloring among F-free graphs, where $F \in \{P_8, S_{2,2,1}, S_{3,1,1}\}$ and explicitly determine all such obstructions. This extends the results of Kamiński and Pstrucha [Discr. Appl. Math. 261, 2019] who proved that there is only a finite number of P_7-free minimal obstructions to C_5-coloring, and of Dębski et al. [ISAAC 2022 Proc.] who showed that the triangle is the unique $S_{2,1,1}$-free minimal obstruction to C_5-coloring.

We complement our results with a construction of an infinite family of minimal obstructions to C_5-coloring, which are simultaneously P_{13}-free and $S_{2,2,2}$-free. We also discuss infinite families of F-free minimal obstructions to H-coloring for other graphs H.

2012 ACM Subject Classification Mathematics of computing → Graph enumeration; Mathematics of computing → Graph coloring; Mathematics of computing → Graph algorithms

Keywords and phrases graph homomorphism, critical graphs, hereditary graph classes

Digital Object Identifier 10.4230/LIPIcs.MFCS.2024.55

Related Version *Full Version*: `https://arxiv.org/abs/2404.11704` [12]

Supplementary Material
Software: `https://github.com/JorikJooken/ObstructionsForC5Coloring` [22]
archived at `swh:1:dir:3bc6333a056d56efa4e99fd55e2061c149fe8b47`

Funding *Jan Goedgebeur*: Jan Goedgebeur is supported by Internal Funds of KU Leuven and an FWO grant with grant number G0AGX24N.
Jorik Jooken: Jorik Jooken is supported by an FWO grant with grant number 1222524N.
Karolina Okrasa: Karolina Okrasa is supported by a Polish National Science Centre grant no. 2021/41/N/ST6/01507.

Paweł Rzążewski: Paweł Rzążewski is supported by a Polish National Science Centre grant no. 2018/31/D/ST6/00062.

Acknowledgements The authors are grateful to Rémi Di Guardia for fruitful and inspiring discussions that led to this project. This project itself was initiated at Dagstuhl Seminar 22481 "Vertex Partitioning in Graphs: From Structure to Algorithms." We are grateful to the organizers and other participants for a productive atmosphere.

1 Introduction

Out of a great number of interesting and elegant graph problems, the notion of graph coloring is, arguably, among the most popular and well-studied ones, not only from combinatorial, but also from algorithmic point of view. For an integer $k \geq 1$, a *k-coloring* of a graph G is a function $c : V(G) \to \{1, \ldots, k\}$ such that for every edge $uv \in E(G)$ it holds that $c(u) \neq c(v)$. For a fixed integer $k \geq 1$, the k-Coloring problem is a computational problem in which an instance is a graph G and we ask whether there exists a k-coloring of G.

The k-Coloring problem is known to be polynomial-time solvable if $k \leq 2$ and NP-complete for larger values of k. Still, even if $k \geq 3$, it is often possible to obtain polynomial-time algorithms that solve k-Coloring if we somehow restrict the class of input instances, for example, to perfect graphs [14, 15], bounded-treewidth graphs [1] or intersection graphs of geometric objects [11].

Observe that these example classes are *hereditary*, i.e., closed under deleting vertices. Such a property is very useful in algorithm design, as it combines well with standard algorithmic techniques, like branching or divide-&-conquer. Therefore, if we want to study some computational problem in a restricted graph class $\mathcal{G}$, choosing $\mathcal{G}$ to be hereditary appears to be reasonable. For a fixed graph F, we say that a graph G is *F-free* if it does not contain F as an induced subgraph. If $\mathcal{F}$ is a family of graphs, we say that a graph G is *$\mathcal{F}$-free* if G is F-free for every $F \in \mathcal{F}$. Each hereditary class of graphs can be characterized by a (possibly infinite) family $\mathcal{F}$ of forbidden induced subgraphs.

Coloring hereditary graph classes. As a first step towards understanding the complexity of k-Coloring in hereditary classes it is natural to consider classes defined by a single induced subgraph F. If F contains a cycle or a vertex of degree at least 3, it follows from the classical results by Emden-Weinert [9], Holyer [20], and Leven and Galil [25] that for every $k \geq 3$, k-Coloring remains NP-complete when restricted to F-free graphs. Thus we are left with the case that F is a *linear forest*, i.e., every component of F is a path.

However, if F is a linear forest, the situation becomes more complicated. For simplicity, let us focus on the case when F is connected, i.e., F is a path on t vertices, denoted by P_t. Then, k-Coloring is polynomial-time-solvable in P_t-free graphs if $t \leq 5$, or if $(k, t) \in \{(3, 6), (3, 7), (4, 6)\}$ [21, 30, 2, 18]. On the other hand, for any $k \geq 4$, the k-Coloring problem is NP-complete in P_t-free graphs for all other values of t [21]. The complexity of the remaining cases, i.e., 3-Coloring of P_t-free graphs where $t \geq 8$, remains unknown: we do not know polynomial-time algorithms nor any hardness proofs. The general belief is that all these cases are in fact tractable, which is supported by the existence of a quasipolynomial-time algorithm for 3-Coloring in P_t-free graphs, for every fixed t [29]. For the summary of the results on the complexity of k-Coloring P_t-free graphs see Figure 1. Let us remark that there are also some results for disconnected forbidden linear forests [24, 6].

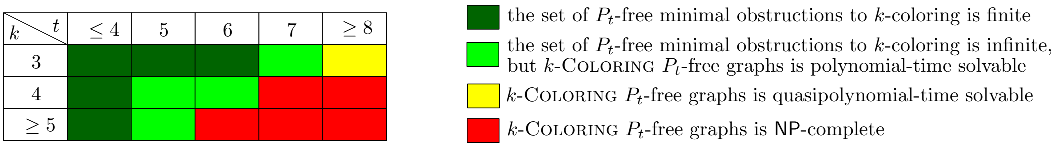

Figure 1 The complexity of k-Coloring P_t-free graphs.

Minimal obstructions to k-coloring. One can look at k-Coloring from another, purely combinatorial perspective. Instead of asking whether a graph G admits a k-coloring, we can ask whether it contains a *dual object*, i.e., some structure that forces the chromatic number to be at least $k+1$. For example, 2-Coloring can be equivalently expressed as a question whether a graph contains an odd cycle. As another example, k-Coloring restricted to perfect graphs is equivalent to the question of the existence of a $(k+1)$-clique, i.e., the complete graph on $k+1$ vertices, denoted by K_{k+1}.

In other words, odd cycles are *minimal non-2-colorable graphs* and $(k+1)$-cliques are *minimal non-k-colorable perfect graphs* (where minimality is defined with respect to the induced subgraph relation). Formally, if a graph G is not k-colorable, but every proper induced subgraph of it is k-colorable, we say that G is *vertex-$(k+1)$-critical* or is a *minimal obstruction to k-coloring*. We denote by $\mathsf{Obstructions}(k)$ the set of all minimal obstructions to k-coloring. Note that $\mathsf{Obstructions}(k)$ naturally forms a family of dual objects – a graph is k-colorable if and only if it does not contain any graph from $\mathsf{Obstructions}(k)$ as an induced subgraph.

Suppose that, for some k, there is a polynomial-time algorithm Alg_k that takes as an input a graph G and answers whether it contains any graph from $\mathsf{Obstructions}(k)$ as an induced subgraph (i.e., whether G is *not* $\mathsf{Obstructions}(k)$-free). From the discussion above it follows that the existence of Alg_k yields a polynomial-time algorithm for k-Coloring. Thus it is unlikely that Alg_k exists for any $k \geq 3$. However, it is still possible when we restrict the input graphs to a certain class $\mathcal{G}$ (like perfect graphs in the example above). Recall that we are interested in the case that $\mathcal{G} = F$-free, where F is a path. Let us denote such a restriction Alg_k to F-free graphs by $\mathsf{Alg}_{k,F}$.

Note that the existence of $\mathsf{Alg}_{k,F}$ is trivial if ($\mathsf{Obstructions}(k) \cap F$-free) is finite; indeed, brute force works in this case. This line of arguments allows us to further refine cases that are polynomial-time solvable: into pairs (k, F), where ($\mathsf{Obstructions}(k) \cap F$-free) is finite, and the others. Recall that the algorithm for k-Coloring obtained for the former ones is able to produce a *negative certificate*: a small (constant-size) witness that the input graph is *not* k-colorable. We refer the reader to the survey of McConnell et al. [27] for more information about certifying algorithms.

It turns out that we can fully characterize all pairs (k, F) for which $\mathsf{Obstructions}(k) \cap F$-free is finite. It is well-known that P_4-free graphs (also known as *cographs*) are perfect and thus the only minimal obstruction to k-coloring is the $(k+1)$-clique. Bruce et al. [3] proved that there is a finite number of minimal obstructions to 3-coloring among P_5-free graphs. The result was later extended by Chudnovsky et al. [4] who showed that the family of P_6-free minimal obstructions to 3-coloring is is also finite, and that this is no longer true among P_7-free graphs (and thus for P_t-free graphs for every $t \geq 7$). If the number of colors is larger, things get more difficult faster: Hoàng et al. [19] showed that for each $k \geq 4$ there exists an infinite family of P_5-free minimal obstructions to k-coloring. See also Figure 1.

H-coloring F-free graphs and minimal obstructions to H-colorings. Graph colorings can be seen as a special case of *graph homomorphisms*. For graphs G and H, an *H-coloring* of G is a function $c : V(G) \to V(H)$ such that for every edge $uv \in E(G)$ it holds that $c(u)c(v) \in E(H)$. The graph H is usually called the *target graph*. It is straightforward to verify that homomorphisms from G to the k-clique, are in one-to-one correspondence to k-colorings of G. For this reason one often refers to the vertices of H as *colors*.

For a fixed graph H, by H-Coloring we denote the computational problem that takes as an input a graph G and asks whether G admits a homomorphism to H. The complexity dichotomy for H-Coloring was proven by Hell and Nešetřil [16]: the problem is polynomial-time solvable if H is bipartite, and NP-complete otherwise.

The complexity landscape of H-Coloring in F-free graphs for non-complete target graphs is far from being fully understood. Chudnovsky et al. [5] proved that if H is an odd cycle on at least 5 vertices, then H-Coloring is polynomial-time solvable in P_9-free graphs; they also showed a number of hardness results for more general variants of the homomorphism problem. Feder and Hell [10] and Dębski et al. [8] studied the case when H is an *odd wheel*, i.e., an odd cycle with universal vertex added. The most general algorithmic results were provided by Okrasa and Rzążewski [28] who showed that

- **(OR1)** if H does not contain C_4 as a subgraph, then H-Coloring can be solved in quasi-polynomial time in P_t-free graphs for any fixed t (note that a better running time here would also mean progress for 3-Coloring P_t-free graphs),
- **(OR2)** if H is of girth at least 5, then H-Coloring can be solved in subexponential time in F-free graphs, where F is any fixed *subdivided claw*, i.e., any graph obtained from the three-leaf star by subdividing edges.

While these are not polynomial-time algorithms, no NP-hardness proofs for these cases are known either. To complete the picture, from [28] it also follows that if H is a so-called *projective core* that contains C_4 as a subgraph, then there exists a t such that H-Coloring is NP-complete in P_t-free graphs (and thus also in graphs excluding some fixed subdivided claw). Furthermore, the hardness reductions even exclude any subexponential-time algorithms for these cases, assuming the Exponential Time Hypothesis (ETH). Let us skip the definition of a projective core, as it is quite technical and not really relevant for this paper. However, it is worth pointing out that almost all graphs are projective cores [26, 17].

Since we are interested in a finer classification of polynomial-time-solvable cases, we should be looking at pairs (H, F) of graphs such that the H-Coloring problem is *not* known to be NP-complete in F-free graphs. From the discussion above it follows that there are two natural families of such pairs to consider:

- **(i)** when H does not contain C_4 as a subgraph and F is a path,
- **(ii)** when H is of girth at least 5 and F is a subdivided claw.

It is straightforward to generalize the notion of minimal obstructions to the setting of H-colorings. A graph G is called a *minimal obstruction to H-coloring* if there is no H-coloring of G, but every proper induced subgraph of G can be H-colored.

The area of minimal obstructions to H-coloring is rather unexplored. In the setting of (i), Kamiński and Pstrucha [23] showed that for any $t \geq 5$, there are finitely many minimal obstructions to C_{t-2}-coloring among P_t-free graphs.[1] In particular, the family of P_7-free

[1] While in [23] the authors consider minimality with respect to the *subgraph* relation, it is not hard to observe that bounded number of subgraph-minimal obstructions is equivalent to the bounded number of induced-subgraph-minimal obstructions.

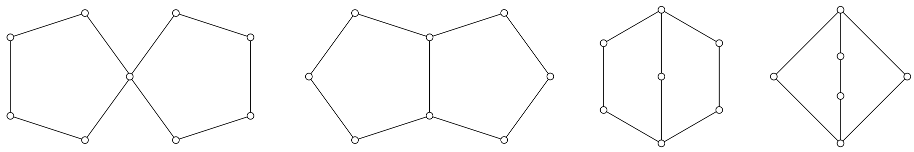

Figure 2 Graphs Q_1, Q_2, Q_3, and Q_4 (left to right).

minimal obstructions to C_5-coloring is finite. In the setting of (ii), Dębski et al. [8] showed that the triangle is the only minimal obstruction to C_5-coloring among graphs that exclude the *fork*, i.e., the graph obtained from the three-leaf star by subdividing one edge once.

Our contribution. As our first result, we show the following strengthening of the result of Kamiński and Pstrucha.

▶ **Theorem 1.** *There are 19 minimal obstructions to C_5-coloring among P_8-free graphs.*

Let us sketch the proof of Theorem 1. Note that K_3 is a minimal obstruction for C_5-coloring, so from now on we focus on graphs that are $\{P_8, K_3\}$-free. For $i \in \{1, 2, 3, 4\}$, by Q_i we denote the graph obtained from two copies of C_5 by identifying an i-vertex subpath of one cycle with an i-vertex subpath of the other one, see Figure 2. In the proof we separately consider minimal obstructions that contain some Q_i as an induced subgraph, and those that are $\{Q_1, Q_2, Q_3, Q_4\}$-free.

The intuition behind this is as follows. Notice that if G contains an induced 5-vertex cycle, the vertices of this cycle must be mapped to the vertices of C_5 bijectively, respecting the ordering along the cycle. Consequently, if G contains some Q_i, the colorings of the vertices in Q_i are somehow restricted. Combining several Q_i's we might impose some contradictory constraints and thus build a graph that is not C_5-colorable. However, as each Q_i already contains quite long induced paths, we might hope that by combining several Q_i we are either forced to create an induced P_8 (if we add only few edges between different Q_i's) or K_3 (if we add too many such edges). Thus the possibilities of building non-C_5-colorable graphs using this approach are somehow limited. It turns out that this intuition is correct: there are 18 graphs that are $\{P_8, K_3\}$-free and contain Q_i, for some $i \in \{1, 2, 3, 4\}$, as an induced subgraph. Together with K_3, they are shown in Figure 3. This part of the proof is computer-aided.

For the second step, we assume that our graph does not contain any Q_i, i.e., we consider graphs that are $\{P_8, K_3, Q_1, Q_2, Q_3, Q_4\}$-free. We show that such graphs are always C_5-colorable. Consequently, each minimal obstruction to C_5-coloring (and, in general, every graph that is not C_5-colorable) was discovered in step 1.

Before we discuss the second result, let us introduce the notation for subdivided claws. For integers $a, b, c \geq 1$, by $S_{a,b,c}$ we denote the graph obtained from the three-leaf star by subdividing each edge, respectively, $a-1$, $b-1$, and $c-1$ times. Equivalently, $S_{a,b,c}$ is obtained from three paths $P_{a+1}, P_{b+1}, P_{c+1}$ by identifying one of their endpoints.

As our second result we show the following extension of the result of Dębski et al. [8].

▶ **Theorem 2.** *There are 3 minimal obstructions to C_5-coloring among $S_{2,2,1}$-free graphs, and 5 minimal obstructions to C_5-coloring among $S_{3,1,1}$-free graphs.*

These graphs are shown in Figure 4. The proof is similar to the proof of Theorem 1. First, we consider minimal obstructions that contain an induced C_5 and, using the computer search, we show that there is only a finite number of them. Then, we show that graphs that

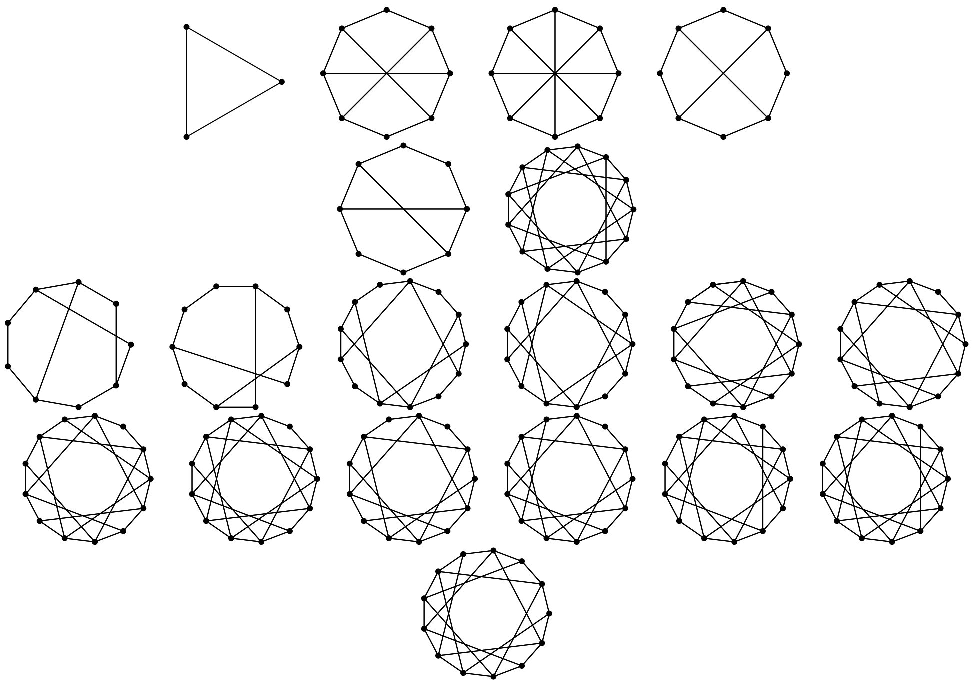

Figure 3 All P_8-free minimal obstructions to C_5-coloring. The graphs in the first row are P_6-free. The graphs in the second row are P_7-free, but not P_6-free. All other graphs are P_8-free, but not P_7-free.

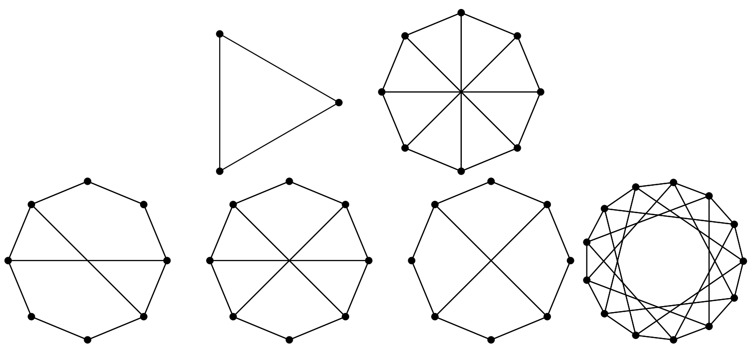

Figure 4 All $S_{2,2,1}$-free and all $S_{3,1,1}$-free minimal obstructions to C_5-coloring. The graphs in the first row are both $S_{2,2,1}$-free and $S_{3,1,1}$-free, whereas the first three graphs in the second row are $S_{3,1,1}$-free, but not $S_{2,2,1}$-free and the last graph in the second row is $S_{2,2,1}$-free, but not $S_{3,1,1}$-free.

exclude K_3 (as it is a minimal obstruction by itself), C_5, and also one of $S_{2,2,1}, S_{3,1,1}$, are either bipartite or are "blown-up cycles" – in both cases C_5-colorability is straightforward to show.

We complement these results with a construction of an infinite family of minimal obstructions to C_5-coloring.

▶ **Theorem 3.** *There is an infinite family of minimal obstructions to C_5-coloring, which simultaneously exclude P_{13}, $S_{2,2,2}$, $S_{5,5,1}$, $S_{11,1,1}$, and $S_{8,2,1}$ as an induced subgraph.*

The construction from Theorem 3 is obtained by generalizing the infinite family of P_7-free minimal obstructions to 3-coloring, provided by Chudnovsky et al. [4]. The idea can be further generalized. Let the *odd girth* of H be the length of a shortest odd cycle in H (and keep it undefined for bipartite graphs).

▶ **Theorem 4.** *Let $q \geq 3$ be odd, and let H be a graph of odd girth q that does not contain C_4 as a subgraph. There is an infinite family of minimal obstructions to H-coloring that are $\{P_{3q-1}, S_{2,2,2}, S_{3(q-1)/2,3(q-1)/2,1}\}$-free.*

Note that Theorem 4 gives a bound for every graph H that was discussed in (OR1) and (OR2). An astute reader might notice that applying Theorem 4 for $H = C_5$, i.e., $q = 5$, yields a family of obstructions that are in particular P_{14}-free and $S_{6,6,1}$-free, which does not match the bounds from Theorem 3. Indeed, obtaining the refined result from Theorem 3 requires some additional work, which again uses a mixture of combinatorial observations and computer search.

Organization of the paper. In Section 2 we introduce some notation and preliminary observations. In Section 3 we explain the algorithm that is later used to generate minimal obstructions. In Sections 4 and 5 we present, respectively, the overview of the proofs of Theorems 1 and 2 In Section 6 we provide constructions of infinite families of graphs that are then used to prove Theorems 3 and 4.

We refer the interested reader to the full version of this paper [12] for the proofs that were omitted due to space limitations (marked by (♠)), and for implementation details of our algorithms and how they were tested for correctness.

2 Preliminaries

For an integer $n \geq 1$ we denote by $[n]$ the set $\{1, \ldots, n\}$, and by $[n]_0$ the set $[n] \cup \{0\}$. For a graph $G = (V, E)$ and a vertex set $U \subseteq V$, the graph $G[U]$ denotes the subgraph of G induced by U. The graph $G - U$ denotes $G[V(G) \setminus U]$. The set $N_G(u)$ denotes the neighborhood of vertex u in the graph G. For $U \subseteq V(G)$ we define $N_G(U) = \bigcup_{u \in U} N_G(u) \setminus U$. If the graph G is clear from the context, we omit the subscript and write $N(u)$ and $N(U)$.

If there exists an H-coloring of G, we denote this fact by $G \to H$. It is straightforward to verify that if $G \to H$, then $\mathsf{odd\text{-}girth}(G) \geq \mathsf{odd\text{-}girth}(H)$. In particular, K_3 has no C_5-coloring and is actually a minimal obstruction to C_5-coloring. Consequently, every other minimal obstruction to C_5-coloring is K_3-free.

For any two graphs G and H such that G is H-colorable, the graph $\mathsf{hull}(G, H)$ denotes the graph with vertex set $V(G)$ and edge set $\{uv : u, v \in V(G)$ and for every H-coloring c of G, we have $c(u)c(v) \in E(H)\}$. Note that $\mathsf{hull}(G, H)$ is a supergraph of G that is H-colorable and that for every induced subgraph G' of G we have $E(\mathsf{hull}(G', H)) \subseteq E(\mathsf{hull}(G, H))$. Note that $\mathsf{hull}(G, H)$ might contain some induced subgraphs that do not appear in G.

3 Generating F-free minimal obstructions to H-coloring

In this section we describe an algorithm that can be used to generate all F-free minimal obstructions to H-coloring. We emphasize that this approach is robust in the sense that it does not assume that $H = C_5$ and F is a path or a subdivided claw, as needed for Theorems 1 and 2.

Throughout the section graphs H and F are fixed. We will use the term *minimal obstruction* for *minimal obstruction to H-coloring*. The algorithm takes as an input "the current candidate graph" I that it tries to extend to a minimal obstruction by adding a new vertex x and some edges between x and $V(I)$. In particular, the algorithm can be used to generate all F-free minimal obstructions by choosing I as the single-vertex graph.

This algorithm is similar to the algorithm for k-Coloring by [13], but we formulate it in a more general way. In case there are infinitely many minimal obstructions, the generation algorithm will never terminate. If there are only finitely many minimal obstructions, then the algorithm might still not terminate, since the prunning rules might not be strong enough. However, if the algorithm terminates, it is guaranteed that there are only finitely many minimal obstructions and that the algorithm outputs all of them.

Let us explain the algorithm, see also the pseudocode in Algorithm 1.

Algorithm 1 Expand.

Constants : target graph H, forbidden graph F
Input : current graph I
Output : exhaustive list of F-free minimal obstructions to H-coloring

if *I is F-free* **and** *not generated before* **then**
  if *I is not H-colorable* **then**
    if *I is a minimal obstruction to H-coloring* **then** **output** I
else
  if *I contains comparable vertices (u, v)* **then**
    foreach *graph I' obtained from I by adding a new vertex x and edges between x and vertices in $V(I)$ in all possible ways, such that $ux \in E(I')$, but $vx \notin E(\mathsf{hull}(I' - u, H))$* **do**
      `Expand(`I'`)`
  else if *I contains comparable edges $(uv, u'v')$* **then**
    foreach *graph I' obtained from I by adding a new vertex x and edges between x and vertices in $V(I)$ in all possible ways, such that $rx \in E(I')$, but $r'x \notin E(\mathsf{hull}(I' - \{u, v\}, H)))$ for some $r \in \{u, v\}$* **do**
      `Expand(`I'`)`
  else
    foreach *graph I' obtained from I by adding a new vertex x and edges between x and vertices in $V(I)$ in all possible ways* **do**
      `Expand(`I'`)`

It starts from a graph I and recursively expands this graph by adding a vertex and edges between this new vertex and existing vertices in each recursive step. The expansion is based on expansion rules that aim at reducing the search space while ensuring that no minimal obstructions are lost in this operation. For example, if an expansion leads to a graph I' that is not F-free, the recursion can be stopped, because all further expansions of I' will not be F-free either (note that we do not add any edges inside $V(I)$ and that the class of F-free graphs is hereditary). Another way to restrict the search space is based on Lemma 5. This lemma and its proof follow Lemma 5 from [4] concerning k-Coloring, but generalizing it to H-Coloring required some adjustments.

▶ **Lemma 5 (♠).** *Let $G = (V, E)$ be a minimal obstruction to H-coloring and let U and W be two non-empty disjoint vertex subsets of G. Let $J := \mathsf{hull}(G - U, H)$. If there exists a homomorphism ϕ from $G[U]$ to $J[W]$, then there exists a vertex $u \in U$ for which $N_G(u) \setminus U \nsubseteq N_J(\phi(u))$.*

As isomorphism is a special type of a homomorphism, Lemma 5 immediately yields the following corollary.

▶ **Corollary 6.** *Let G be an H-colorable graph that is an induced subgraph of a minimal obstruction G'. Let $U, W \subseteq V(G)$ be two non-empty disjoint vertex subsets and let $J := \mathsf{hull}(G - U, H)$. If there exists an isomorphism $\phi : G[U] \to J[W]$ such that $N_G(u) \setminus U \subseteq N_J(\phi(u))$ for all $u \in U$, then there exists a vertex $x \in V(G') \setminus V(G)$ such that x is adjacent to some vertex $u \in U$, but x is not adjacent to $\phi(u)$ in $\mathsf{hull}(G' - U, H)$ (and thus also not adjacent to $\phi(u)$ in $\mathsf{hull}(G'[V(G) \cup \{x\} \setminus U], H)$).*

Actually, we will only use the restricted version of Corollary 6. In what follows we use the notation and assumptions of the Corollary. In case that $|U| = |W| = 1$, say $U = \{u\}$ and $W = \{w\}$, we call the pair (u, w) *comparable vertices*. In case that $G[U]$ and $J[W]$ are both isomorphic to K_2 and, say, $U = \{u, u'\}$ and $W = \{w, w'\}$, we call the pair (uu', ww') *comparable edges*. The algorithm concentrates on finding comparable vertices and edges for computational reasons.

We refer the interested reader to the full version of this paper [12] for additional details about the efficient implementation of this algorithm, independent correctness verifications and sanity checks.

4 Minimal obstructions to C_5-coloring with no long paths

In this section we still only discuss C_5-colorings, thus we will keep writing *minimal obstructions* for *minimal obstructions for C_5-coloring*.

The algorithm from Section 3 was implemented for $H = C_5$ (the source code is made publicly available at [22]). We used the algorithm, combined with some purely combinatorial observations, to generate an exhaustive list of P_t-free minimal obstructions, where $t \leq 8$; see also Figure 3. The minimal obstructions can also be obtained from the database of interesting graphs at the *House of Graphs* [7] by searching for the keywords "minimal obstruction to C5-coloring".

4.1 P_t-free minimal obstructions for $t \in \{6, 7\}$

As a warm-up, let us reprove the result of Kamiński and Pstrucha [23] (in a slightly stronger form, as they did not provide the explicit list of minimal obstructions). It will also serve as a demonstration of the way how the algorithm from 3 is intended to be used.

An exhaustive list for $t \leq 6$ can be obtained by running the algorithm from Section 3 with parameters $(I = K_1, H = C_5, F = P_6)$.[2] This leads to the following observation.

▶ **Observation 7.** *There are four minimal obstructions for C_5-coloring among P_6-free graphs. All of these obstructions, except for the triangle K_3, are P_6-free and not P_5-free.*

Unfortunately, the same simple strategy already fails for $t = 7$. Indeed, the algorithm as presented in Section 3 does not terminate after running for several hours after calling it with parameters $(I = K_1, H = C_5, F = P_7)$. However, with relatively small adjustments, the algorithm is able to produce an exhaustive list of minimal obstructions in a few seconds.

The first adjustment has to do with the order in which the expansion rules are used. Note that the order in which the algorithm checks whether it can find comparable vertices and comparable edges does not affect the correctness of the algorithm, but it might affect

[2] Let us remark that it is a simple exercise to find this list by hand.

whether the algorithm terminates or not. For example, it could happen that by expanding in order to get rid of a pair of comparable vertices, a new pair of comparable vertices is introduced (and this could continue indefinitely). By applying a different expansion rule first, this can be avoided sometimes. For $F = P_7$, the algorithm was run by first looking for comparable vertices and then for comparable edges, except when $|V(I)| = 10$, in which case the algorithm first looks for comparable edges and then for comparable vertices.

The second adjustment is based on the following observation.

▶ **Observation 8 (♠).** *Every P_7-free minimal obstruction to C_5-coloring, except for the graph K_3, contains the cycle C_5 or the cycle C_7 as an induced subgraph.*

By Observation 8, the set of P_7-free minimal obstructions can be partitioned into three subsets:

(i) the triangle K_3,

(ii) minimal obstructions that are P_7-free but contain C_5 as an induced subgraph,

(iii) minimal obstructions that are P_7-free but contain C_7 as an induced subgraph.

Thus, running the algorithm for $(I = C_5, H = C_5, F = P_7)$ and $(I = C_7, H = C_5, F = P_7)$, respectively, we can generate the families (ii) and (iii). This yields the following result; recall that the finiteness of the family of minimal obstructions was already shown by Kamiński and Pstrucha [23].

▶ **Observation 9.** *There are six P_7-free minimal obstructions to C_5-coloring. Two of these obstructions are P_7-free, but not P_6-free.*

4.2 P_8-free minimal obstructions

This section is devoted to the proof of Theorem 1, which we restate below (see also Figure 3).

▶ **Theorem 1.** *There are 19 minimal obstructions to C_5-coloring among P_8-free graphs.*

Similarly to the P_7-free case, the proof uses the algorithm from Section 3, but this time it requires a lot more purely combinatorial insights. For $i \in [4]$, let Q_i be the graph obtained from two disjoint copies of C_5 by identifying i pairs of consecutive corresponding vertices of the cycles (see Figure 2). Let G be a P_8-free minimal obstruction; we aim to understand the structure of G and show that is must be one of 19 graphs in Figure 3. We split the reasoning into two cases: first, we assume that G contains Q_i, for some $i \in [4]$, as an induced subgraph. Then, in the second case, we assume that G is $\{Q_1, Q_2, Q_3, Q_4\}$-free.

Case 1: G contains Q_i for some $i \in [4]$ as an induced subgraph. We deal with this case using the algorithm from Section 3. The algorithm terminates in a few minutes when it is called with the parameters $(I = Q_i, H = C_5, F = P_8)$ for all $i \in [4]$. All minimal obstructions obtained this way are listed in Figure 3.

Case 2: G does not contain Q_i for any $i \in [4]$. As K_3 is a minimal obstruction, from now on we can assume that G is $\{P_8, K_3, Q_1, Q_2, Q_3, Q_4\}$-free. We aim to show that all such graphs are C_5-colorable, i.e., the list obtained in Case 1, plus the triangle, is exhaustive.

▶ **Lemma 10 (♠).** *Let G be a $\{P_8, K_3, Q_1, Q_2, Q_3, Q_4\}$-free graph. Then G is C_5-colorable.*

5 Minimal obstructions to C_5-coloring with no long subdivided claws

This section is devoted to the proof of Theorem 2.

▶ **Theorem 2.** *There are 3 minimal obstructions to C_5-coloring among $S_{2,2,1}$-free graphs, and 5 minimal obstructions to C_5-coloring among $S_{3,1,1}$-free graphs.*

The minimal obstructions are shown in Figure 4 and can also be obtained from the database of interesting graphs at the *House of Graphs* [7] by searching for the keywords "S221-free minimal obstruction to C5-coloring" and "S311-free minimal obstruction to C5-coloring", respectively. As in this section the target graph is always C_5, we will keep writing *minimal obstructions* for *minimal obstructions to C_5-coloring*. We proceed similarly as in the proof of Theorem 1. Let G be an $S_{2,2,1}$-free (respectively, $S_{3,1,1}$-free) minimal obstruction. We again consider two cases.

Case 1: G contains C_5 as an induced subgraph. This case is solved using the algorithm from Section 3. The algorithm is called with the parameters $(I = C_5, H = C_5, F = S_{2,2,1})$ and then with parameters $(I = C_5, H = C_5, F = S_{3,1,1})$. Both calls terminate, returning a finite list of minimal obstructions.

Case 2: G does not contain C_5 as an induced subgraph. Similarly as in the proof of Theorem 1, note that K_3 is a minimal $\{F, C_5\}$-free obstruction for $F \in \{S_{2,2,1}, S_{3,1,1}\}$. Thus, from now on, we assume that G is K_3-free and prove that there are no more minimal obstructions satisfying this case, i.e., the following result.

▶ **Lemma 11 (♠).** *Let $F \in \{S_{2,2,1}, S_{3,1,1}\}$ and let G be a $\{F, C_5, K_3\}$-free graph. Then G is C_5-colorable.*

Combining the cases, we obtain the statement of Theorem 2.

6 An infinite family of minimal obstructions

In this section we construct infinite families of graphs that will be later used to prove Theorems 3 and 4. The construction is a generalization of the one designed for 3-Coloring [4]; the authors attribute it to Pokrovskiy.

The construction. For every odd $q \geq 3$ and every $p \geq 1$, let $G_{q,p}$ be the graph on vertex set $[qp-3]_0$ (all arithmetic operations on $[qp-3]_0$ here are done modulo $qp-2$), such that for every $i \in [qp-3]_0$ it holds that

$$N(i) = \{i-1, i+1\} \cup \{i + qj - 1 \mid j \in [p-1]\}.$$

To simplify the arguments, we partition $V(G_{q,p})$ into q sets $V_s = \{i \mid i = s \mod q\}$, where $s \in [q-1]_0$. Next observation follows immediately from the definition of $E(G_{q,p})$.

▶ **Observation 12.** *Let $q \geq 3$ be odd, and let $ij \in E(G_{q,p})$ be such that $i \in V_s$ for some $s \in [q-1]_0$. If $i < j$, then $j \in V_{s-1}$, or $j = i+1$, or $i = 0$ and $j = qp-3$. Analogously, if $j > i$, then $j \in V_{s+1}$, or $j = i-1$, or $i = qp-3$ and $j = 0$.*

We now show that graphs $G_{q,p}$ are minimal obstructions to H-coloring for a rich family of graphs H.

▶ **Lemma 13 (♠).** *Let $q \geq 3$ be an odd integer, and let H be graph of odd girth q that does not contain C_4 as a subgraph. For every $p \geq 1$ the graph $G_{q,p}$ is a minimal obstruction to H-coloring.*

Excluded induced subgraphs of $G_{q,p}$. Now let us show an auxiliary lemma that will be helpful in analyzing induced subgraphs that (do not) appear in $G_{q,p}$. In particular, it implies that in order to prove that for each p, the graph $G_{q,p}$ is F-free for some graph F, it is sufficient to show that this statement holds for some small values of p.

▶ **Lemma 14.** *Let q be a fixed constant and F be a graph. Let $p \geq |V(F)| + 2$. If $G_{q,p-1}$ is F-free, then $G_{q,p}$ is F-free.*

Proof. Assume otherwise, and let $U \subseteq V(G_{q,p})$ induce a copy of F in $G_{q,p}$, in particular $|V(F)| = |U|$. Since $|V(F)| < qp-2$, we can assume without loss of generality that the vertex $qp-3$ does not belong to U. Since $|V(G_{q,p})| = qp-2 > q(|V(F)|+1)$, there exist $q+1$ consecutive vertices $\ell, \ldots, \ell+q$ that do not belong to U. Define $R = \{j \in U \mid j > \ell + q\}$, $R' = \{j - q \mid j \in R\}$, and let $L = U \setminus R$.

Now consider $U' = L \cup R'$, and note that $\ell \notin U'$. It is straightforward to verify that $U' \subseteq V(G_{q,p-1})$. We will show that U' induces a copy of F in $V(G_{q,p-1})$. Since this is a contradiction with our assumption, we then conclude that $G_{q,p}$ is F-free.

Let $i, j \in U'$, let $s \in [q-1]_0$ be such that $i \in V_s$. Assume without loss of generality that $i < j$. Note that it is enough to show that

- if $i, j < \ell$, then $ij \in E(G_{q,p-1})$ if and only if $ij \in E(G_{q,p})$,
- if $i, j > \ell$, then $ij \in E(G_{q,p-1})$ if and only if $(i+q)(j+q) \in E(G_{q,p})$,
- if $i < \ell < j$, then $ij \in E(G_{q,p-1})$ if and only if $i(j+q) \in E(G_{q,p})$.

The first item is straightforward.

For the second item, by Observation 12 we have that $ij \in E(G_{q,p-1})$ if and only if $j = i+1$ or $j \in V_{s-1}$. The first is equivalent to $j+q = (i+q)+1$, the latter is equivalent to $j+q \in V_{s-1}$. Hence again using Observation 12 we obtain that $ij \in E(G_{q,p-1})$ if and only if $(i+q)(j+q) \in E(G_{q,p})$.

For the last item, note that $i < \ell < j$ implies $i \in L$ and $j \in R'$. If $ij \in E(G_{q,p-1})$, then by Observation 12, either $j = i+1$ or $v_j \in V_{s-1}$. Since $i < \ell < j$, the first case is not possible. In the second case, if $v_j \in V_{s-1}$, then $j+q \in V_{s-1}$, so $i(j+q) \in E(G_{q,p})$. On the other hand, if $i(j+q) \in E(G_{q,p})$, then, since $i < \ell < j$, it cannot happen that $j+q = i+1$. If $j+q \in V_{s-1}$ then $j \in V_{s-1}$, so we conclude that $ij \in E(G_{q,p-1})$. That concludes the proof. ◀

The power of Lemma 14 is that in order to show that $G_{q,p}$ is F-free for *every* p, it is sufficient to prove it for a finite (and small) set of graphs. This is encapsulated in the following, immediate corollary.

▶ **Corollary 15.** *Let q be a fixed constant and F be a graph. If $G_{q,p}$ is F-free for every $p \leq |V(F)| + 1$, then $G_{q,p}$ is F-free for every p.*

Consequently, for every fixed q and F, Corollary 15 reduces the problem of showing that $G_{q,p}$ is F-free to a constant-size task that can be tackled with a computer.

Proof of Theorem 4 and Theorem 3. Now let us analyze what induced paths and subdivided claws appear in $G_{q,p}$. We start with showing that for every odd $q \geq 3$ and every $p \geq 1$ the graph $G_{q,p}$ is qK_2-free, i.e., they exclude an induced matching on q edges. Here, an induced matching is a set of edges that are not only pairwise disjoint, but also non-adjacent.

▶ **Lemma 16 (♠).** *Let $q \geq 3$ be an odd integer. For every $p \geq 1$ the graph $G_{q,p}$ is qK_2-free.*

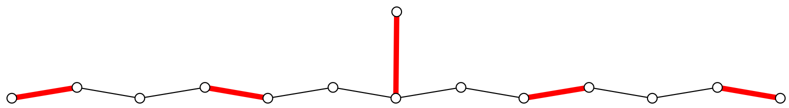

Figure 5 If a graph contains $S_{3(q-1)/2,3(q-1)/2,1}$ as an induced subgraph, then it also contains qK_2.

Let us remark that Lemma 16 is best possible, i.e., if p is large enough, then $G_{q,p}$ contains $(q-1)K_2$ as an induced subgraph. We do not prove it, as later, in Lemma 19, we will show a stronger result. Let us turn our attention to induced paths and subdivided claws that do not appear in $G_{q,p}$. In particular, using Lemma 16 we can exclude the existence of long paths and claws (see Figure 5).

▶ **Lemma 17** (♠). *Let $q \geq 3$ be an odd integer. For every $p \geq 1$ the graph $G_{q,p}$ is $\{P_{3q-1}, S_{2,2,2}, S_{3(q-1)/2,3(q-1)/2,1}\}$-free.*

Now, as an immediate consequence of Lemma 13 and Lemma 17, we obtain Theorem 4, which we restate below.

▶ **Theorem 4.** *Let $q \geq 3$ be odd, and let H be a graph of odd girth q that does not contain C_4 as a subgraph. There is an infinite family of minimal obstructions to H-coloring that are $\{P_{3q-1}, S_{2,2,2}, S_{3(q-1)/2,3(q-1)/2,1}\}$-free.*

Note that for $H = C_5$, i.e., for $q = 5$, Theorem 4 shows that the constructed graphs are in particular P_{14}-free and $S_{6,6,1}$-free. It turns out that they are actually P_{13}-free and $S_{5,5,1}$-free (and also exclude some other subdivided claws). Here we will make use of Corollary 15, combined with computer search.

Let us start with analyzing the length of a longest induced path in $G_{5,p}$. Thus we are interested in applying Corollary 15 to the case $q = 5$ and $F = P_{13}$. Actually, we can even exclude $F = P_{10} + P_2$, i.e., the graph with two components: one isomorphic to P_{10} and the other isomorphic to P_2. Note that $(P_{10} + P_2)$-free graphs are in particular P_{13}-free. Furthermore, the graph $G_{5,p}$ excludes the following subdivided claws: $S_{5,5,1}$, $S_{11,1,1}$, and $S_{8,2,1}$. These results, together with Lemma 13, give us Theorem 3.

▶ **Theorem 3.** *There is an infinite family of minimal obstructions to C_5-coloring, which simultaneously exclude P_{13}, $S_{2,2,2}$, $S_{5,5,1}$, $S_{11,1,1}$, and $S_{8,2,1}$ as an induced subgraph.*

Longest induced paths in $G_{q,p}$. Theorem 3, and the fact that from the result of Chudnovsky et al. [4] it follows that for every p, the graph $G_{3,p}$ is P_7-free (while Lemma 17 only gives P_8-freeness), suggest that the bound on the length of a longest induced path given by Theorem 4 is not optimal also in the other cases. This evidence suggests the following conjecture.

▶ **Conjecture 18.** *Let $q \geq 3$ be odd. For every $p \geq 1$, the graph $G_{q,p}$ is P_{3q-2}-free.*

We conclude this section by showing that the bound from Conjecture 18, if true, is best possible.

▶ **Lemma 19** (♠). *Let $q \geq 3$ be odd. If $p \geq 2q+1$, then $G_{q,p}$ contains P_{3q-3} as an induced subgraph.*

References

1 Stefan Arnborg and Andrzej Proskurowski. Linear time algorithms for NP-hard problems restricted to partial k-trees. *Discret. Appl. Math.*, 23(1):11–24, 1989. doi:10.1016/0166-218X(89)90031-0.

2 Flavia Bonomo, Maria Chudnovsky, Peter Maceli, Oliver Schaudt, Maya Stein, and Mingxian Zhong. Three-coloring and list three-coloring of graphs without induced paths on seven vertices. *Comb.*, 38(4):779–801, 2018. doi:10.1007/s00493-017-3553-8.

3 Daniel Bruce, Chính T. Hoàng, and Joe Sawada. A certifying algorithm for 3-colorability of P_5-free graphs. In Yingfei Dong, Ding-Zhu Du, and Oscar H. Ibarra, editors, *Algorithms and Computation, 20th International Symposium, ISAAC 2009, Honolulu, Hawaii, USA, December 16-18, 2009. Proceedings*, volume 5878 of *Lecture Notes in Computer Science*, pages 594–604. Springer, 2009. doi:10.1007/978-3-642-10631-6_61.

4 Maria Chudnovsky, Jan Goedgebeur, Oliver Schaudt, and Mingxian Zhong. Obstructions for three-coloring graphs without induced paths on six vertices. *J. Comb. Theory, Ser. B*, 140:45–83, 2020. doi:10.1016/j.jctb.2019.04.006.

5 Maria Chudnovsky, Shenwei Huang, Pawel Rzążewski, Sophie Spirkl, and Mingxian Zhong. Complexity of C_k-coloring in hereditary classes of graphs. *Inf. Comput.*, 292:105015, 2023. doi:10.1016/J.IC.2023.105015.

6 Maria Chudnovsky, Shenwei Huang, Sophie Spirkl, and Mingxian Zhong. List 3-Coloring graphs with no induced $P_6 + rP_3$. *Algorithmica*, 83(1):216–251, 2021. doi:10.1007/S00453-020-00754-Y.

7 Kris Coolsaet, Sven D'hondt, and Jan Goedgebeur. House of Graphs 2.0: A database of interesting graphs and more. *Discret. Appl. Math.*, 325:97–107, 2023. doi:10.1016/J.DAM.2022.10.013.

8 Michał Dębski, Zbigniew Lonc, Karolina Okrasa, Marta Piecyk, and Paweł Rzążewski. Computing Homomorphisms in Hereditary Graph Classes: The Peculiar Case of the 5-Wheel and Graphs with No Long Claws. In Sang Won Bae and Heejin Park, editors, *33rd International Symposium on Algorithms and Computation (ISAAC 2022)*, volume 248 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 14:1–14:16, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ISAAC.2022.14.

9 Thomas Emden-Weinert, Stefan Hougardy, and Bernd Kreuter. Uniquely colourable graphs and the hardness of colouring graphs of large girth. *Comb. Probab. Comput.*, 7(4):375–386, 1998. URL: http://journals.cambridge.org/action/displayAbstract?aid=46667.

10 Tomás Feder and Pavol Hell. Edge list homomorphisms. Unpublished manuscript, available at http://theory.stanford.edu/~tomas/edgelist.ps.

11 M. R. Garey, David S. Johnson, Gerald L. Miller, and Christos H. Papadimitriou. The complexity of coloring circular arcs and chords. *SIAM J. Algebraic Discret. Methods*, 1(2):216–227, 1980. doi:10.1137/0601025.

12 Jan Goedgebeur, Jorik Jooken, Karolina Okrasa, Paweł Rzążewski, and Oliver Schaudt. Minimal obstructions to C_5-coloring in hereditary graph classes. *CoRR*, abs/2404.11704, 2024. arXiv:2404.11704.

13 Jan Goedgebeur and Oliver Schaudt. Exhaustive generation of k-critical H-free graphs. *J. Graph Theory*, 87(2):188–207, 2018. doi:10.1002/JGT.22151.

14 Martin Grötschel, László Lovász, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Comb.*, 1(2):169–197, 1981. doi:10.1007/BF02579273.

15 Martin Grötschel, László Lovász, and Alexander Schrijver. Corrigendum to our paper "The ellipsoid method and its consequences in combinatorial optimization". *Comb.*, 4(4):291–295, 1984. doi:10.1007/BF02579139.

16 Pavol Hell and Jaroslav Nešetřil. On the complexity of H-coloring. *J. Comb. Theory, Ser. B*, 48(1):92–110, 1990. doi:10.1016/0095-8956(90)90132-J.

17 Pavol Hell and Jaroslav Nešetřil. The core of a graph. *Discret. Math.*, 109(1-3):117–126, 1992. `doi:10.1016/0012-365X(92)90282-K`.

18 Chính T. Hoàng, Marcin Kamiński, Vadim V. Lozin, Joe Sawada, and Xiao Shu. Deciding k-colorability of P_5-free graphs in polynomial time. *Algorithmica*, 57(1):74–81, 2010. `doi:10.1007/s00453-008-9197-8`.

19 Chính T. Hoàng, Brian Moore, Daniel Recoskie, Joe Sawada, and Martin Vatshelle. Constructions of k-critical P_5-free graphs. *Discret. Appl. Math.*, 182:91–98, 2015. `doi:10.1016/j.dam.2014.06.007`.

20 Ian Holyer. The NP-completeness of edge-coloring. *SIAM J. Comput.*, 10:718–720, 1981.

21 Shenwei Huang. Improved complexity results on k-coloring P_t-free graphs. *Eur. J. Comb.*, 51:336–346, 2016. `doi:10.1016/j.ejc.2015.06.005`.

22 Jorik Jooken. Source code of algorithm `Expand`. `https://github.com/JorikJooken/ObstructionsForC5Coloring`, 2024.

23 Marcin Kamiński and Anna Pstrucha. Certifying coloring algorithms for graphs without long induced paths. *Discret. Appl. Math.*, 261:258–267, 2019. `doi:10.1016/j.dam.2018.09.031`.

24 Tereza Klimošová, Josef Malík, Tomás Masařík, Jana Novotná, Daniël Paulusma, and Veronika Slívová. Colouring $(P_r + P_s)$-free graphs. *Algorithmica*, 82(7):1833–1858, 2020. `doi:10.1007/S00453-020-00675-W`.

25 Daniel Leven and Zvi Galil. NP-completeness of finding the chromatic index of regular graphs. *J. Algorithms*, 4(1):35–44, 1983. `doi:10.1016/0196-6774(83)90032-9`.

26 Tomasz Łuczak and Jaroslav Nešetřil. Note on projective graphs. *J. Graph Theory*, 47(2):81–86, 2004. `doi:10.1002/JGT.20017`.

27 Ross M. McConnell, Kurt Mehlhorn, Stefan Näher, and Pascal Schweitzer. Certifying algorithms. *Comput. Sci. Rev.*, 5(2):119–161, 2011. `doi:10.1016/J.COSREV.2010.09.009`.

28 Karolina Okrasa and Paweł Rzążewski. Complexity of the list homomorphism problem in hereditary graph classes. In Markus Bläser and Benjamin Monmege, editors, *38th International Symposium on Theoretical Aspects of Computer Science, STACS 2021, March 16-19, 2021, Saarbrücken, Germany (Virtual Conference)*, volume 187 of *LIPIcs*, pages 54:1–54:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. `doi:10.4230/LIPIcs.STACS.2021.54`.

29 Marcin Pilipczuk, Michał Pilipczuk, and Paweł Rzążewski. Quasi-polynomial-time algorithm for Independent Set in P_t-free graphs via shrinking the space of induced paths. In Hung Viet Le and Valerie King, editors, *4th Symposium on Simplicity in Algorithms, SOSA 2021, Virtual Conference, January 11-12, 2021*, pages 204–209. SIAM, 2021. `doi:10.1137/1.9781611976496.23`.

30 Sophie Spirkl, Maria Chudnovsky, and Mingxian Zhong. Four-coloring P_6-free graphs. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1239–1256. SIAM, 2019. `doi:10.1137/1.9781611975482.76`.