FINDING LARGE H-COLORABLE SUBGRAPHS IN HEREDITARY GRAPH CLASSES *

MARIA CHUDNOVSKY†, JASON KING‡, MICHAŁ PILIPCZUK§, PAWEŁ RZĄŻEWSKI¶, AND SOPHIE SPIRKL‖

Abstract. We study the Max Partial H-Coloring problem: given a graph G, find the largest induced subgraph of G that admits a homomorphism into H, where H is a fixed pattern graph without loops. Note that when H is a complete graph on k vertices, the problem reduces to finding the largest induced k-colorable subgraph, which for $k = 2$ is equivalent (by complementation) to Odd Cycle Transversal.

We prove that for every fixed pattern graph H without loops, Max Partial H-Coloring can be solved:

- in $\{P_5, F\}$-free graphs in polynomial time, whenever F is a threshold graph;
- in $\{P_5, \text{bull}\}$-free graphs in polynomial time;
- in P_5-free graphs in time $n^{\mathcal{O}(\omega(G))}$;
- in $\{P_6, \text{1-subdivided claw}\}$-free graphs in time $n^{\mathcal{O}(\omega(G)^3)}$.

Here, n is the number of vertices of the input graph G and $\omega(G)$ is the maximum size of a clique in G. Furthermore, by combining our algorithms for P_5-free and for $\{P_6, \text{1-subdivided claw}\}$-free graphs with a simple branching procedure, we obtain subexponential-time algorithms for Max Partial H-Coloring in these classes of graphs.

Finally, we show that even a restricted variant of Max Partial H-Coloring is NP-hard in the considered subclasses of P_5-free graphs, if we allow loops on H.

Key words. odd cycle tranversal, graph homomorphism, P_5-free graphs

AMS subject classifications. 05C15, 05C85, 68R10

1. Introduction.

Many computational graph problems that are (NP-)hard in general become tractable in restricted classes of input graphs. In this work we are interested in *hereditary* graph classes, or equivalently classes defined by forbidding induced subgraphs. For a set of graphs $\mathcal{F}$, we say that a graph G is $\mathcal{F}$*-free* if G does not contain any induced subgraph isomorphic to a graph from $\mathcal{F}$. By forbidding different sets $\mathcal{F}$ we obtain graph classes with various structural properties, which can be used in the algorithmic context. This highlights an interesting interplay between structural graph theory and algorithm design.

Perhaps the best known example of this paradigm is the case of the Maximum Independent Set problem: given a graph G, find the largest set of pairwise non-adjacent vertices in G. It is known that the problem is NP-hard on F-free graphs

*The extended abstract of this paper was presented on the conference ESA 2020 [10].
Submitted to the editors on 16 Sep 2020.

Funding: MC: This material is based upon work supported in part by the U. S. Army Research Office under grant number W911NF-16-1-0404, and by NSF grant DMS-1763817.
MP: This work is a part of project TOTAL that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 677651).
PRz: Supported by Polish National Science Centre grant no. 2018/31/D/ST6/00062.
SS: This material is based upon work supported by the National Science Foundation under Award No. DMS1802201.

†Princeton University, Princeton, NJ 08544(mchudnov@math.princeton.edu)

‡Princeton University, Princeton, NJ 08544 (jtking@princeton.edu)

§Institute of Informatics, University of Warsaw, Poland (michal.pilipczuk@mimuw.edu.pl)

¶Faculty of Mathematics and Information Science, Warsaw University of Technology, Poland, and Institute of Informatics, University of Warsaw, Poland, (p.rzazewski@mini.pw.edu.pl)

‖Princeton University, Princeton, NJ 08544. Current address: University of Waterloo, Waterloo, Ontario, Canada. (sspirkl@math.princeton.edu)

unless F is a forest whose every component is a path or a subdivided claw [2]; here, the *claw* is the star with 3 leaves. However, the remaining cases, when F is a *subdivided claw forest*, remain largely unexplored despite significant effort. Polynomial-time algorithms have been given for P_5-free graphs [34], P_6-free graphs [28], claw-free graphs [37, 41], and fork-free graphs [3, 35]. While the complexity status in all the other cases remains open, it has been observed that relaxing the goal of polynomial-time solvability leads to positive results in a larger generality. For instance, for every $t \in \mathbb{N}$, Maximum Independent Set can be solved in time $n^{\mathcal{O}(\log^2 n)}$ in P_t-free graphs [24, 40]. Moreover, if F is fixed a subdivided claw forest, then the problem can be solved in time $2^{\mathcal{O}(n^{8/9})}$ [12, 13]. The existence of such *quasipolynomial-time* and *subexponential-time* algorithms for F-free graphs is excluded under the Exponential Time Hypothesis whenever F is not a subdivided claw forest (see e.g. the discussion in [38]), which shows a qualitative difference between the negative and the potentially positive cases.

The abovementioned positive results use a variety of structural techniques related to the considered hereditary graph classes, for instance: the concept of *Gyárfás path* that gives useful separators in P_t-free graphs [4, 7, 13], the dynamic programming approach based on potential maximal cliques [34, 28], or structural properties of claw-free and fork-free graphs that relate them to line graphs [35, 37, 41]. Some of these techniques can be used to give algorithms for related problems, which can be expressed as looking for the largest (in terms of the number of vertices) induced subgraph satisfying a fixed property. For Maximum Independent Set this property is being edgeless, but for instance the property of being acyclic corresponds to the Maximum Induced Forest problem, which by complementation is equivalent to Feedback Vertex Set. Work in this direction so far focused on properties that imply bounded treewidth [1, 23, 25] or, more generally, that imply sparsity [38].

A different class of problems that admits an interesting complexity landscape on hereditary graphs classes are coloring problems. For fixed $k \in \mathbb{N}$, the k-Coloring problem asks whether the input graph admits a proper coloring with k colors. For every $k \geq 3$, the problem is NP-hard on F-free graphs unless F is a forest of paths (a *linear forest*) [26]. The classification of the remaining cases is more advanced than in the case of Maximum Independent Set, but not yet complete. On one hand, Hoàng et al. [32] showed that for every fixed k, k-Coloring is polynomial-time solvable on P_5-free graphs. On the other hand, the problem becomes NP-hard already on P_6-free graphs for all $k \geq 5$ [33]. The cases $k = 3$ and $k = 4$ turn out to be very interesting. 4-Coloring is polynomial-time solvable on P_6-free graphs [17] and NP-hard in P_7-free graphs [33]. While there is a polynomial-time algorithm for 3-Coloring in P_7-free graphs [5], the complexity status in P_t-free graphs for $t \geq 8$ remains open. However, relaxing the goal again leads to positive results in a wider generality: for every $t \in \mathbb{N}$, there is a quasipolynomial-time algorithm with running time $n^{\mathcal{O}(\log^2 n)}$ for 3-Coloring in P_t-free graphs [40], and there is also a polynomial-time algorithm that given a 3-colorable P_t-free graph outputs its proper coloring with $\mathcal{O}(t)$ colors [15].

We are interested in using the toolbox developed for coloring problems in P_t-free graphs to the setting of finding maximum induced subgraphs with certain properties. Specifically, consider the following Maximum Induced k-Colorable Subgraph problem: given a graph G, find the largest induced subgraph of G that admits a proper coloring with k colors. While this problem clearly generalizes k-Coloring, for $k = 1$ it boils down to Maximum Independent Set. For $k = 2$ it can be expressed as

MAXIMUM INDUCED BIPARTITE SUBGRAPH, which by complementation is equivalent to the well-studied ODD CYCLE TRANSVERSAL problem: find the smallest subset of vertices that intersects all odd cycles in a given graph. While polynomial-time solvability of ODD CYCLE TRANSVERSAL on P_4-free graphs (also known as *cographs*) follows from the fact that these graphs have bounded cliquewidth (see [18]), it is known that the problem is NP-hard in P_6-free graphs [21]. The complexity status of ODD CYCLE TRANSVERSAL in P_5-free graphs remains open [11, Problem 4.4]: resolving this question was the original motivation of our work. Let us point out that the complexity of MAXIMUM INDUCED k-COLORABLE SUBGRAPH in hereditary graph classes was considered already in the 1980s [42].

Our contribution. Following the work of Groenland et al. [27], we work with a very general form of coloring problems, defined through homomorphisms. For graphs G and H, a *homomorphism* from G to H, or an H-*coloring* of G, is a function $\phi\colon V(G) \to V(H)$ such that for every edge uv in G, we have $\phi(u)\phi(v) \in E(H)$. We study the MAX PARTIAL H-COLORING problem defined as follows: given a graph G, find the largest induced subgraph of G that admits an H-coloring. Note that if H is the complete graph on k vertices, then an H-coloring is simply a proper coloring with k colors, hence this formulation generalizes the MAXIMUM INDUCED k-COLORABLE SUBGRAPH problem. We will always assume that the pattern graph H does not have loops, hence an H-coloring is always a proper coloring with $|V(H)|$ colors.

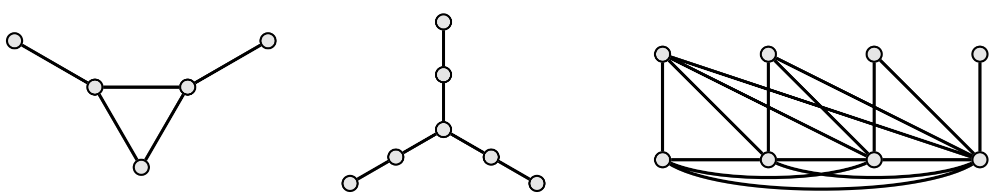

FIG. 1. *A bull, a 1-subdivided claw, and an example threshold graph.*

Fix a pattern graph H without loops. We prove that MAX PARTIAL H-COLORING can be solved:

(R1) in $\{P_5, F\}$-free graphs in polynomial time, whenever F is a threshold graph;
(R2) in $\{P_5, \text{bull}\}$-free graphs in polynomial time;
(R3) in P_5-free graphs in time $n^{\mathcal{O}(\omega(G))}$; and
(R4) in $\{P_6, \text{1-subdivided claw}\}$-free graphs in time $n^{\mathcal{O}(\omega(G)^3)}$.

Here, n is the number of vertices of the input graph G and $\omega(G)$ is the size of the maximum clique in G. Also, recall that a graph G is a *threshold graph* if $V(G)$ can be partitioned into an independent set A and a clique B such that for each $a, a' \in A$, we have either $N(a) \supseteq N(a')$ or $N(a) \subseteq N(a')$. There is also a characterization via forbidden induced subgraphs: threshold graphs are exactly $\{2P_2, C_4, P_4\}$-free graphs, where $2P_2$ is an induced matching of size 2. Figure 1 depicts a bull, a 1-subdivided claw, and an example threshold graph.

Further, we observe that by employing a simple branching strategy, an $n^{\mathcal{O}(\omega(G)^\alpha)}$-time algorithm for MAX PARTIAL H-COLORING in $\mathcal{F}$-free graphs can be used to give also a subexponential-time algorithm in this setting, with running time $n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$. Thus, results (R3) and (R4) imply that for every fixed irreflexive H, the MAX PARTIAL H-COLORING problem can be solved in time $n^{\mathcal{O}(\sqrt{n})}$ in P_5-free graphs and in time $n^{\mathcal{O}(n^{3/4})}$ in $\{P_6, \text{1-subdivided claw}\}$-free graphs. This in particular applies to the ODD

Cycle Transversal problem. We note here that Dabrowski et al. [21] proved that Odd Cycle Transversal in $\{P_6, K_4\}$-free graphs is NP-hard and does not admit a subexponential-time algorithm under the Exponential Time Hypothesis. Thus, it is unlikely that any of our algorithmic results — the $n^{\mathcal{O}(\omega(G))}$-time algorithm and the $n^{\mathcal{O}(\sqrt{n})}$-time algorithm — can be extended from P_5-free graphs to P_6-free graphs.

All our algorithms work in a weighted setting, where instead of just maximizing the size of the domain of an H-coloring, we maximize its total *revenue*, where for each pair $(u, v) \in V(G) \times V(H)$ we have a prescribed revenue yielded by sending u to v. This setting allows encoding a broader range of coloring problems. For instance, list variants can be expressed by giving negative revenues for forbidden assignments (see e.g. [29, 39]). Also, our algorithms work in a slightly larger generality than stated above, see Section 5, Section 6, and Section 7 for precise statements.

Finally, we investigate the possibility of extending our algorithmic results to pattern graphs with possible loops. We show an example of a graph H with loops, for which Max Partial H-Coloring is NP-hard and admits no subexponential-time algorithm under the ETH even in very restricted subclasses of P_5-free graphs, including $\{P_5, \text{bull}\}$-free graphs. This shows that whether the pattern graph is allowed to have loops has a major impact on the complexity of the problem.

Our techniques. The key element of our approach is a branching procedure that, given an instance (G, rev) of Max Partial H-Coloring, where rev is the revenue function, produces a relatively small set of instances Π such that solving (G, rev) reduces to solving all the instances in Π. Moreover, every instance $(G', \mathsf{rev}') \in \Pi$ is simpler in the following sense: either it is an instance of Max Partial H'-Coloring for H' being a proper induced subgraph of H (hence it can be solved by induction on $|V(H)|$), or for any connected graph F on at least two vertices, G' is F-free provided we assume G is $F^{\bullet\!-\!\circ}$-free. Here $F^{\bullet\!-\!\circ}$ is the graph obtained from F by adding a universal vertex y and a degree-1 vertex x adjacent only to y. In particular we have $\omega(G') < \omega(G)$, so applying the branching procedure exhaustively in a recursion scheme yields a recursion tree of depth bounded by $\omega(G)$. Now, for results (R3) and (R4) we respectively have $|\Pi| \leq n^{\mathcal{O}(1)}$ and $|\Pi| \leq n^{\mathcal{O}(\omega(G)^2)}$, giving bounds of $n^{\mathcal{O}(\omega(G))}$ and $n^{\mathcal{O}(\omega(G)^3)}$ on the total size of the recursion tree and on the overall time complexity.

For result (R1) we apply the branching procedure not exhaustively, but a constant number of times: if the original graph G is $\{P_5, F\}$-free for some threshold graph F, it suffices to apply the branching procedure $\mathcal{O}(|V(F)|)$ times to reduce the original instances to a set of edgeless instances, which can be solved trivially. As $\mathcal{O}(|V(F)|) = \mathcal{O}(1)$, this gives recursion tree of polynomial size, and hence a polynomial-time complexity due always having $|\Pi| \leq n^{\mathcal{O}(1)}$ in this setting. For result (R2), we show that two applications of the branching procedure reduce the input instance to a polynomial number of instances that are P_4-free, which can be solved in polynomial time due to P_4-free graphs (also known as *cographs*) having cliquewidth at most 2. However, these applications are interleaved with a reduction to the case of *prime graphs* — graphs with no non-trivial modules — which we achieve using dynamic programming on the modular decomposition of the input graph. This is in order to apply some results on the structure of prime bull-free graphs [14, 16], so that P_4-freeness is achieved at the end.

Let us briefly discuss the key branching procedure. The first step is finding a useful dominating structure that we call a *monitor*: a subset of vertices M of a connected graph G is a monitor if for every connected component C of $G - M$, there is a vertex in M that is complete to C. We prove that in a connected P_6-free graph there is always a monitor that is the closed neighborhood of a set of at most three vertices. After finding such a monitor $N[X]$ for $|X| \leq 3$, we perform a structural analysis of the graph centered around the set X. This analysis shows that there exists a subset of $\mathcal{O}(|V(H)|)$ vertices such that after guessing this subset and the H-coloring on it, the instance can be partitioned into several separate subinstances, each of which has a strictly smaller clique number. This structural analysis, and in particular the way the separation of subinstances is achieved, is inspired by the polynomial-time algorithm of Hoàng et al. [32] for k-Coloring in P_5-free graphs.

Other related work. We remark that very recently and independently of us, Brettell et al. [9] proved that for every fixed $s, t \in \mathbb{N}$, the class of $\{K_t, sK_1+P_5\}$-free graphs has bounded mim-width. Here, *mim-width* is a graph parameter that is less restrictive than cliquewidth, but the important aspect is that a wide range of vertex-partitioning problems, including the Max Partial H-Coloring problem considered in this work, can be solved in polynomial time on every class of graphs where the mim-width is universally bounded and a corresponding decomposition can be computed efficiently. The result of Brettell et al. thus shows that in P_5-free graphs, the mim-width is bounded by a function of the clique number. This gives an $n^{f(\omega(G))}$-time algorithm for Max Partial H-Coloring in P_5-free graphs (for fixed H), for some function f. However, the proof presented in [9] gives only an exponential upper bound on the function f, which in particular does not imply the existence of a subexponential-time algorithm. To compare, our reasoning leads to an $n^{\mathcal{O}(\omega(G))}$-time algorithm and a subexponential-time algorithm with complexity $n^{\mathcal{O}(\sqrt{n})}$.

We remark that the techniques used by Brettell et al. [9] also rely on revisiting the approach of Hoàng et al. [32], and they similarly observe that this approach can be used to apply induction based on the clique number of the graph.

Organization. After setting up notation and basic definition in Section 2 and proving an auxiliary combinatorial result about P_6-free graphs in Section 3, we provide the key technical lemma (Lemma 4.1) in Section 4. This lemma captures a single branching step of our algorithms. In Section 5 we derive results (R3) and (R4). Section 6 and Section 7 are devoted to the proofs of results (R1) and (R2), respectively. In Section 8 we show that allowing loops in H may result in an NP-hard problem even in restricted subclasses of P_5-free graphs. We conclude in Section 9 by discussing directions of further research.

2. Preliminaries.

Graphs. For a graph G, the vertex and edge sets of G are denoted by $V(G)$ and $E(G)$, respectively. The *open neighborhood* of a vertex u is the set $N_G(u) \coloneqq \{v \colon uv \in E(G)\}$, while the *closed neighborhood* is $N_G[u] \coloneqq N_G(u) \cup \{u\}$. This notation is extended to sets of vertices: for $X \subseteq V(G)$, we set $N_G[X] \coloneqq \bigcup_{u \in X} N_G[u]$ and $N_G(X) \coloneqq N_G[X] \setminus X$. We may omit the subscript if the graph G is clear from the context. By C_t, P_t, and K_t we respectively denote the cycle, the path, and the complete graph on t vertices.

The *clique number* $\omega(G)$ is the size of the largest clique in a graph G. A clique K in G is *maximal* if no proper superset of K is a clique.

For $s, t \in \mathbb{N}$, the *Ramsey number* of s and t is the smallest integer k such that every graph on k vertices contains either a clique of size s or an independent set of size t. It is well-known that the Ramsey number of s and t is bounded from above by $\binom{s+t-2}{s-1}$, hence we will denote $\mathsf{Ramsey}(s,t) \coloneqq \binom{s+t-2}{s-1}$.

For a graph G and $A \subseteq V(G)$, by $G[A]$ we denote the subgraph of G induced by

A. We write $G - A := G[V(G) \setminus A]$. We say that F is an *induced subgraph* of G if there is $A \subseteq V(G)$ such that $G[A]$ is isomorphic to F; this containment is *proper* if in addition $A \neq V(G)$. For a family of graphs $\mathcal{F}$, a graph G is *$\mathcal{F}$-free* if G does not contain any induced subgraph from $\mathcal{F}$. If $\mathcal{F} = \{H\}$, then we may speak about *H-free* graphs as well.

If G is a graph and $A \subseteq V(G)$ is a subset of vertices, then a vertex $u \notin A$ is *complete* to A if u is adjacent to all the vertices of A, and u is *anti-complete* to A if u has no neighbors in A. We will use the following simple claim several times.

LEMMA 2.1. *Suppose G is a graph, A is a subset of its vertices such that $G[A]$ is connected, and $u \notin A$ is a vertex that is neither complete nor anti-complete to A in G. Then there are vertices $a, b \in X$ such that $u - a - b$ is an induced P_3 in G.*

Proof. Since u is neither complete nor anticomplete to A, both the sets $A \cap N(u)$ and $A \setminus N(u)$ are non-empty. As A is connected, there exist $a \in A \cap N(u)$ and $b \in A \setminus N(u)$ such that a and b are adjacent. Now $u - a - b$ is the desired induced P_3. ∎

For a graph F, by $F^{\bullet}$ we denote the graph obtained from F by adding a *universal vertex*: a vertex adjacent to all the other vertices. Similarly, by $F^{\bullet\!-\!\circ}$ we denote the graph obtained from F by adding first an isolated vertex, say x, and then a universal vertex, say y. Note that thus y is adjacent to all the other vertices of $F^{\bullet\!-\!\circ}$, while x is adjacent only to y.

H-colorings. For graphs H and G, a function $\phi\colon V(G) \to V(H)$ is a *homomorphism* from G to H if for every $uv \in E(G)$, we also have $\phi(u)\phi(v) \in E(H)$. Note that a homomorphism from G to the complete graph K_t is nothing else than a proper coloring of G with t colors. Therefore, a homomorphism from G to H will be also called an *H-coloring* of G, and we will refer to vertices of H as colors. Note that we will always assume that H is a simple graph without loops, so no two adjacent vertices of G can be mapped by a homomorphism to the same vertex of H. To stress this, we will call such H an *irreflexive pattern graph*.

A *partial homomorphism* from G to H, or a *partial H-coloring* of G, is a partial function $\phi\colon V(G) \rightharpoonup V(H)$ that is a homomorphism from $G[\mathsf{dom}\,\phi]$ to H, where $\mathsf{dom}\,\phi$ denotes the domain of ϕ.

Suppose that with graphs G and H we associate a *revenue function* $\mathsf{rev}\colon V(G) \times V(H) \to \mathbb{R}$. Then the *revenue* of a partial H-coloring ϕ is defined as

$$\mathsf{rev}(\phi) := \sum_{u \in \mathsf{dom}\,\phi} \mathsf{rev}(u, \phi(u)).$$

In other words, for $u \in V(G)$ and $v \in V(H)$, $\mathsf{rev}(u, v)$ denotes the revenue yielded by assigning $\phi(u) := v$.

We now define the main problem studied in this work. In the following, we consider the graph H fixed.

MAX PARTIAL H-COLORING
Input: Graph G and a revenue function $\mathsf{rev}\colon V(G) \times V(H) \to \mathbb{R}$
Output: A partial H-coloring ϕ of G that maximizes $\mathsf{rev}(\phi)$

An *instance* of the MAX PARTIAL H-COLORING problem is a pair (G, rev) as above. A *solution* to an instance (G, rev) is a partial H-coloring of G, and it is *optimum* if it maximizes $\mathsf{rev}(\phi)$ among solutions. By $\mathrm{OPT}(G, \mathsf{rev})$ we denote the maximum possible revenue of a solution to the instance (G, rev).

Let us note one aspect that will be used later on. Observe that in revenue functions we allow negative revenues for some assignments. However, if we are interested in maximizing the total revenue, there is no point in using such assignments: if $u \in \mathsf{dom}\,\phi$ and $\mathsf{rev}(u, \phi(u)) < 0$, then just removing u from the domain of ϕ increases the revenue. Thus, optimal solutions never use assignments with negative revenues. Note that this feature can be used to model list versions of partial coloring problems.

3. Monitors in P_6-free graphs.

In this section we prove an auxiliary result about finding useful separators in P_6-free graphs. The desired property is expressed in the following definition.

DEFINITION 3.1. *Let G be a connected graph. A subset of vertices $M \subseteq V(G)$ is a* monitor *in G if for every connected component C of $G - M$, there exists a vertex $w \in M$ that is complete to C.*

Let us note the following property of monitors.

LEMMA 3.2. *If M is a monitor in a connected graph G, then every maximal clique in G intersects M. In particular, $\omega(G - M) < \omega(G)$.*

Proof. If K is a clique in $G - M$, then K has to be entirely contained in some connected component C of $G - M$. Since M is a monitor, there exists $w \in M$ that is complete to C. Then $K \cup \{w\}$ is also a clique in G, hence K cannot be a maximal clique in G. □

We now prove that in P_6-free graphs we can always find easily describable monitors.

LEMMA 3.3. *Let G be a connected P_6-free graph. Then for every $u \in V(G)$ there exists a subset of vertices X such that $u \in X$, $|X| \leq 3$, $G[X]$ is a path whose one endpoint is u, and $N_G[X]$ is a monitor in G.*

Lemma 3.3 follows immediately from the following statement applied for $t = 6$.

LEMMA 3.4. *Let $t \in \{4, 5, 6\}$, G be a connected P_6-free graph, and $u \in V(G)$ be a vertex such that in G there is no induced P_t with u being one of the endpoints. Then there exists a subset X of vertices such that $u \in X$, $|X| \leq t-3$, $G[X]$ is a path whose one endpoint is u, and $N_G[X]$ is a monitor in G.*

Proof. We proceed by induction on t. The base case for $t = 4$ will be proved directly within the analysis.

In the following, by *slabs* we mean connected components of the graph $G - N_G[u]$. We shall say that a vertex $w \in N_G(u)$ is *mixed* on a slab C if w is neither complete nor anti-complete to C. A slab C is *simple* if there exists a vertex $w \in N_G(u)$ that is complete to C, and *difficult* otherwise.

Note that since G is connected, for every difficult slab D there exists some vertex $w \in N_G(u)$ that is mixed on D. Then, by Lemma 2.1, we can find vertices $a, b \in D$ such that $u - w - a - b$ is an induced P_4 in G. If $t = 4$ then no such induced P_4 can exists, so we infer that in this case there are no difficult slabs. Then $N_G[u]$ is a monitor, so we may set $X \coloneqq \{u\}$. This proves the claim for $t = 4$; from now on we assume that $t \geq 5$.

Let us choose a vertex $v \in N_G(u)$ that maximizes the number of difficult slabs on which v is mixed. Suppose there is a difficult slab D' such that v is anti-complete to D'. As we argued, there exists a vertex $v' \in N_G(u)$ such that v' is mixed on D'; clearly $v' \neq v$. By the choice of v, there exists a difficult slab D such that v is mixed on D and v' is anti-complete to D. By applying Lemma 2.1 twice, we find vertices

$a, b \in D$ and $a', b' \in D'$ such that $v - a - b$ and $v' - a' - b'$ are induced P_3s in G. Now, if v and v' were adjacent, then $a - b - v - v' - a' - b'$ would be an induced P_6 in G, a contradiction. Otherwise $a - b - v - u - v' - a' - b'$ is an induced P_7 in G, again a contradiction (see Figure 2).

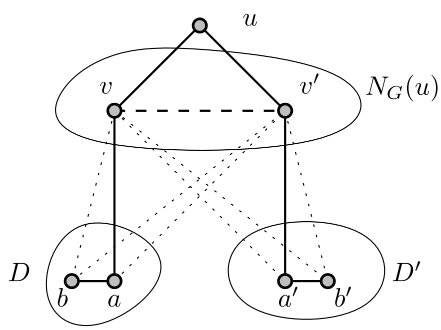

Fig. 2. *The graph G in the proof of Lemma 3.4 when v anti-complete to some difficult slab D'. Dotted lines show non-edges. The edge vv' might be present.*

We conclude that v is mixed on every difficult slab. Let

$$A \coloneqq \{v\} \cup \bigcup_{D\colon \text{difficult slab}} V(D).$$

Then $G[A]$ is connected and P_6-free. Moreover, in $G[A]$ there is no P_{t-1} with one endpoint being v, because otherwise we would be able to extend such an induced P_{t-1} using u, and thus obtain an induced P_t in G with one endpoint being u. Consequently, by induction we find a subset $Y \subseteq A$ such that $|Y| \leq t-4$, $G[Y]$ is a path with one of the endpoints being v, and $N_{G[A]}[Y]$ is a monitor in $G[A]$. Let $X \coloneqq Y \cup \{u\}$. Then $|X| \leq t-3$ and $G[X]$ is a path with u being one of the endpoints.

We verify that $N_G[X]$ is a monitor in G. Consider any connected component C of $G - N_G[X]$. As $N_G[X] \supseteq N_G[u]$, C is contained in some slab D. If D is simple, then by definition there exists a vertex $w \in N_G[u] \subseteq N_G[X]$ that is complete to D, hence also complete to C. Otherwise D is difficult, hence C is a connected component of $G[A] - N_{G[A]}[Y]$. Since $N_{G[A]}[Y]$ is a monitor in $G[A]$, there exists a vertex $w \in N_{G[A]}[Y] \subseteq N_G[X]$ that is complete to C. This completes the proof. ◻

We remark that no statement analogous to Lemma 3.3 may hold for P_7-free graphs, even if from X we only require that $N_G[X]$ intersects all the maximum-size cliques in G (which is implied by the property of being a monitor, see Lemma 3.2). Consider the following example. Let G be a graph obtained from the union of $n+1$ complete graphs $K^{(0)}, \ldots, K^{(n)}$, each on n vertices, by making one vertex from each of the graphs $K^{(1)}, \ldots, K^{(n)}$ adjacent to a different vertex of $K^{(0)}$. Then G is P_7-free, but the minimum size of a set $X \subseteq V(G)$ such that $N_G[X]$ intersects all maximum-size cliques in G is n.

4. Branching. We now present the core branching step that will be used by all our algorithms. This part is inspired by the approach of Hoàng et al. [32]. We will rely on the following two graph families; see Figure 3. For $t \in \mathbb{N}$, the graph S_t is obtained from the star $K_{1,t}$ by subdividing every edge once. Then $L_1 \coloneqq P_3$ and for $t \geq 2$ the graph L_t is obtained from S_t by making all the leaves of S_t pairwise adjacent.

Fig. 3. *Graphs S_4 and L_4.*

Lemma 4.1. *Let H be a fixed irreflexive pattern graph. Suppose we are given integers s, t and an instance (G, rev) of* Max Partial H-Coloring *such that G is connected and $\{P_6, L_s, S_t\}$-free, and the range of $\mathsf{rev}(\cdot)$ contains at least one positive value. Denoting $n \coloneqq |V(G)|$, one can in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ compute a set Π of size $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ such that the following conditions hold:*

(B1) Each element of Π is a pair $((G_1, \mathsf{rev}_1), (G_2, \mathsf{rev}_2))$, where G_1, G_2 are $\{P_6, L_s, S_t\}$-free subgraphs of G satisfying $V(G) = V(G_1) \uplus V(G_2)$. Further, (G_2, rev_2) is an instance of Max Partial H-Coloring, *and (G_1, rev_1) is an instance of* Max Partial H'-Coloring, *where H' is some proper induced subgraph of H (which may be different for different elements of Π).*

(B2) For each $((G_1, \mathsf{rev}_1), (G_2, \mathsf{rev}_2)) \in \Pi$ and every connected graph F on at least two vertices, if G_1 contains an induced F, then G contains an induced $F^{\bullet}$. Moreover, if G_2 contains an induced F, then G contains an induced $F^{\bullet\!-\!\circ}$.

(B3) We have

$$\mathrm{OPT}(G, \mathsf{rev}) = \max \{ \mathrm{OPT}(G_1, \mathsf{rev}_1) + \mathrm{OPT}(G_2, \mathsf{rev}_2) \; : \; ((G_1, \mathsf{rev}_1), (G_2, \mathsf{rev}_2)) \in \Pi \}.$$

Moreover, for any pair $((G_1, \mathsf{rev}_1), (G_2, \mathsf{rev}_2)) \in \Pi$ for which this maximum is reached, and for every pair of optimum solutions ϕ_1 and ϕ_2 to (G_1, rev_1) and (G_2, rev_2), respectively, the function $\phi \coloneqq \phi_1 \cup \phi_2$ is an optimum solution to (G, rev) with $\mathsf{rev}(\phi) = \mathsf{rev}_1(\phi_1) + \mathsf{rev}_2(\phi_2)$.

The remainder of this section is devoted to the proof of Lemma 4.1. We fix the irreflexive pattern graph H and consider an input instance (G, rev). We find it more didactic to first perform an analysis of (G, rev), and only provide the algorithm at the end. Thus, the correctness will be clear from the previous observations.

Let

$$T \coloneqq \{ (x, y) \in V(G) \times V(H) \; : \; \mathsf{rev}(x, y) > 0 \}.$$

By assumption T is nonempty, hence $\mathrm{OPT}(G, \mathsf{rev}) > 0$ and every optimum solution ϕ to (G, rev) has a nonempty domain: it sets $\phi(x) = y$ for some $(x, y) \in T$. Consequently, the final set Π will be obtained by taking the union of sets $\Pi^{x,y}$ for $(x, y) \in T$: when constructing $\Pi^{x,y}$ our goal is to capture all solutions satisfying $\phi(x) = y$. We now focus on constructing $\Pi^{x,y}$, hence we assume that we fix a pair $(x, y) \in T$.

Since G is connected, by Lemma 3.3 there exists $X \subseteq V(G)$ such that $x \in X$, $|X| \leq 3$, $G[X]$ is a path with x being one of the endpoints, and $N[X]$ is a monitor in G. Note that such X can be found in polynomial time by checking all subsets of $V(G) \setminus \{x\}$ of size at most 2. In case $|X| < 3$, we may add arbitrary to X so that $|X| = 3$ and $G[X]$ remains connected; note that this does not spoil the property that $G[X]$ is a monitor. We may also enumerate the vertices of X as $\{x_1, x_2, x_3\}$ so that $x = x_1$ and for each $i \in \{2, 3\}$ there exists $i' < i$ such that x_i and $x_{i'}$ are adjacent.

We partition $V(G) \setminus X$ into A_1, A_2, A_3, A_4 as follows:

$$A_1 \coloneqq N(x_1) \setminus X, \qquad A_2 \coloneqq N(x_2) \setminus (X \cup A_1),$$
$$A_3 \coloneqq N(x_3) \setminus (X \cup A_1 \cup A_2), \quad A_4 \coloneqq V(G) \setminus N[X].$$

Note that $\{A_1, A_2, A_3\}$ is a partition of $N(X)$ (see Figure 4). For $i \in \{1,2,3\}$, denote $A_{>i} \coloneqq \bigcup_{j=i+1}^{4} A_j$ and observe that x_i is complete to A_i and anti-complete to $A_{>i}$. Moreover, we have the following.

Claim 4.2. *Let F be a connected graph. If $G[A_1]$ contains an induced F, then G contains an induced $F^{\bullet}$. If $G[A_i]$ contains an induced F for any $i \in \{2,3,4\}$, then G contains an induced $F^{\bullet\!-\!\circ}$.*

Proof of Claim. For the first assertion observe that if $B \subseteq A_1$ induces F in G, then $B \cup \{x_1\}$ induces $F^{\bullet}$ in G. For the second assertion, consider first the case when $i \in \{2,3\}$. As we argued, there is $i' < i$ such that $x_{i'}$ and x_i are adjacent. Then if $B \subseteq A_i$ induces F in G, then $B \cup \{x_{i'}, x_i\}$ induces $F^{\bullet\!-\!\circ}$ in G.

We are left with justifying the second assertion for $i = 4$. Suppose $B \subseteq A_4$ induces F in G. Since F is connected, B is entirely contained in one connected component C of $G[A_4]$. As $N[X]$ is a monitor in G, there exists a vertex $w \in N[X]$ that is complete to C. As $w \in N[X]$, some $x_{i'} \in X$ is adjacent to w. We now find that $B \cup \{w, x_{i'}\}$ induces $F^{\bullet\!-\!\circ}$ in G. ■

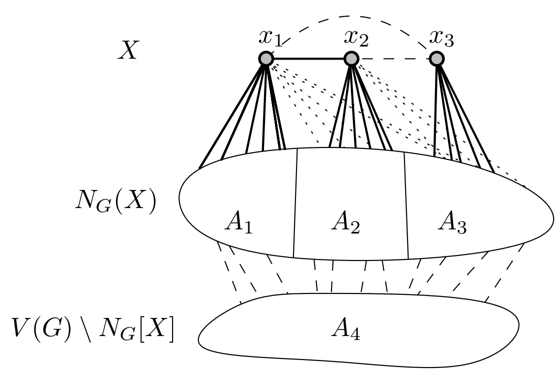

Fig. 4. *The partition on $V(G)$ in the proof of Lemma 4.1. Solid and dotted lines respectively indicate that a vertex is complete or anticomplete to a set. Dashed edges might, but do not have to exist.*

The next claim contains the core combinatorial observation of the proof.

Claim 4.3. *Let ϕ be a solution to the instance (G, rev). Then for every $i \in \{1,2,3\}$ and $v \in V(H)$, there exists a set $S \subseteq A_i$ such that:*

- $|S| < \mathsf{Ramsey}(s,t)$;
- $S \subseteq A_i \cap \phi^{-1}(v)$; *and*
- *every vertex $u \in A_{>i}$ that has a neighbor in $A_i \cap \phi^{-1}(v)$, also has a neighbor in S.*

Proof of Claim. Let S be the smallest set satisfying the second and the third condition, it exists, as these conditions are satisfied by $A_i \cap \phi^{-1}(v)$. Note that since H is irreflexive, it follows that $\phi^{-1}(v)$ is an independent set in G, hence S is independent as well.

Suppose for contradiction that $|S| \geq \mathsf{Ramsey}(s,t)$. By minimality, for every $u \in S$ there exists $u' \in A_{>i}$ such that u is the only neighbor of u' in S. Let $S' \coloneqq \{u' \colon u \in S\}$

(see Figure 5). Since $|S'| \geq \mathsf{Ramsey}(s,t)$, in $G[S']$ we can either find a clique K' of size s or an independent set I' of size t; denote $K \coloneqq \{u \colon u' \in K'\}$ and $I \coloneqq \{u \colon u' \in I'\}$. In the former case, we find that $\{x_i\} \cup K \cup K'$ induces the graph L_s in G, a contradiction. Similarly, in the latter case we have that $\{x_i\} \cup I \cup I'$ induces S_t in G, again a contradiction. This completes the proof of the claim. ■

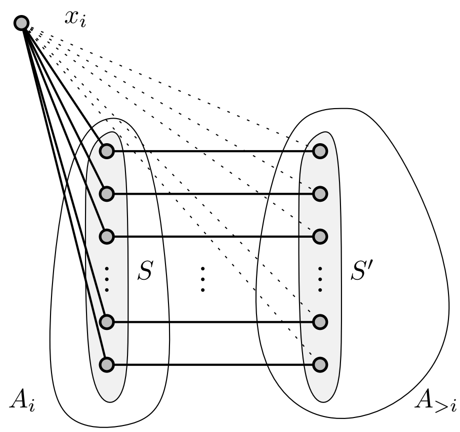

FIG. 5. *Sets S and S' in the proof of Claim 4.3.*

Claim 4.3 suggests the following notion. A *guess* is a function $R \colon V(H) \to 2^{N[X]}$ satisfying the following:

- for each $v \in V(H)$, $R(v)$ is a subset of $N[X]$ such that $|R(v) \cap A_i| < \mathsf{Ramsey}(s,t)$ for all $i \in \{1,2,3\}$;
- sets $R(v)$ are pairwise disjoint for different $v \in V(H)$; and
- $x \in R(y)$.

Let $\mathcal{R}^{x,y}$ be the family of all possible guesses. Note that we add the pair (x,y) in the superscript to signify that the definition of $\mathcal{R}^{x,y}$ depends on (x,y).

CLAIM 4.4. *We have that $|\mathcal{R}^{x,y}| \leq n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ and $\mathcal{R}^{x,y}$ can be enumerated in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$.*

Proof of Claim. For each $v \in V(H)$, the number of choices for $R(v)$ in a guess R is bounded by $2^3 \cdot n^{3 \cdot \mathsf{Ramsey}(s,t)}$: the first factor corresponds to the choice of $R(v) \cap X$, while the second factor bounds the number of choices of $R(v) \cap A_i$ for $i \in \{1,2,3\}$. Since the guess R is determined by choosing $R(v)$ for each $v \in V(H)$ and $|V(H)|$ is considered a constant, the number of different guesses is bounded by $\left(2^3 \cdot n^{3 \cdot \mathsf{Ramsey}(s,t)}\right)^{|V(H)|} = n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$. Clearly, they can be also enumerated in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$. ■

Now, we say that a guess R is *compatible* with a solution ϕ to (G, rev) if the following conditions hold for every $v \in V(H)$:

(C1) $R(v) \subseteq \phi^{-1}(v)$;
(C2) $R(v) \cap X = \phi^{-1}(v) \cap X$; and
(C3) for all $i \in \{1,2,3\}$ and $u \in A_{>i}$, if u has a neighbor in $\phi^{-1}(v) \cap A_i$, then u also has a neighbor in $R(v) \cap A_i$.

The following statement follows immediately from Claim 4.3.

CLAIM 4.5. *For every solution ϕ to the instance (G, rev) which satisfies $\phi(x) = y$, there exists a guess $R \in \mathcal{R}^{x,y}$ that is compatible with ϕ.*

Let us consider a guess $R \in \mathcal{R}^{x,y}$. We define a set $B^R \subseteq V(G) \times V(H)$ of *disallowed pairs* for R as follows. We include a pair $(u,v) \in V(G) \times V(H)$ in B^R if any of the following four conditions holds:

(D1) $u \in X$ and $u \notin R(v)$;
(D2) $u \in R(v')$ for some $v' \in V(H)$ that is different from v;
(D3) u has a neighbor in G that belongs to $R(v')$ for some $v' \in V(H)$ such that $vv' \notin E(H)$; or
(D4) $u \in A_i \setminus R(v)$ for some $i \in \{1,2,3\}$ and there exists $u' \in A_{>i}$ such that $uu' \in E(G)$ and $N_G(u') \cap A_i \cap R(v) = \emptyset$.

Intuitively, B^R contains assignments that contradict the supposition that R is compatible with a considered solution. The fact that $x = x_1$ is complete to A_1 and the assumption $x \in R(y)$ directly yield the following.

Claim 4.6. *For all $u \in A_1$ and $R \in \mathcal{R}^{x,y}$, we have $(u,y) \in B^R$.*

Based on B^R, we define a new revenue functions $\mathsf{rev}^R \colon V(G) \times V(H) \to \mathbb{R}$ as follows:

$$\mathsf{rev}^R(u,v) = \begin{cases} -1 & \text{if } (u,v) \in B^R; \\ \mathsf{rev}(u,v) & \text{otherwise.} \end{cases}$$

The intuition is that if a pair (u,v) is disallowed by R, then we model this in rev^R by assigning negative revenue to the corresponding assignment. This forbids optimum solutions to use this assignment.

We now define a subgraph $G^{x,y}$ of G as follows:

$$V(G^{x,y}) \coloneqq V(G) \text{ and } E(G^{x,y}) \coloneqq \{\, uv \in E(G) \;:\; u,v \in A_i \text{ for some } i \in \{1,2,3,4\} \,\}.$$

In other words, $G^{x,y}$ is obtained from G by removing all edges except those whose both endpoints belong to the same set A_i, for some $i \in \{1,2,3,4\}$.

For every guess $R \in \mathcal{R}^{x,y}$, we may consider a new instance $(G^{x,y}, \mathsf{rev}^R)$ of Max Partial H-Coloring. In the following two claims we establish the relationship between solutions to the instance (G, rev) and solutions to instances $(G^{x,y}, \mathsf{rev}^R)$. The proofs essentially boil down to a verification that all the previous definitions work as expected. In particular, the key point is that the modification of revenues applied when constructing rev^R implies automatic satisfaction of all the constraints associated with edges that were present in G, but got removed in $G^{x,y}$.

Claim 4.7. *For every guess $R \in \mathcal{R}^{x,y}$, every optimum solution ϕ to the instance $(G^{x,y}, \mathsf{rev}^R)$ is also a solution to the instance (G, rev), and moreover $\mathsf{rev}^R(\phi) = \mathsf{rev}(\phi)$.*

Proof of Claim. Recall that ϕ is a solution to (G, rev) if and only if ϕ is a partial H-coloring of G. Hence, we need to prove that for every $uu' \in E(G)$ with $u, u' \in \mathsf{dom}\,\phi$, we have $\phi(u)\phi(u') \in E(H)$. Denote $v \coloneqq \phi(u)$ and $v' \coloneqq \phi(u')$ and suppose for contradiction that $vv' \notin E(H)$. Since ϕ is an optimum solution to $(G^{x,y}, \mathsf{rev}^R)$, we have $\mathsf{rev}^R(u,v) \geq 0$, which implies that $(u,v) \notin B^R$. Similarly $(u',v') \notin B^R$. We now consider cases depending on the alignment of u and u' in G.

If $u, u' \in A_i$ for some $i \in \{1,2,3,4\}$ then $uu' \in E(G^{x,y})$, so the supposition $vv' \notin E(H)$ would contradict the assumption that ϕ is a solution to $(G^{x,y}, \mathsf{rev}^R)$.

Suppose $u \in A_i$ and $u' \in A_j$ for $i,j \in \{1,2,3,4\}$, $i \neq j$; by symmetry, assume $i < j$. As $vv' \notin E(H)$, we infer that u' does not have any neighbors in $R(v)$ in G, for otherwise we would have $(u',v') \in B^R$ by (D3). As $uu' \in E(G)$, $u \in A_i$, and $u' \in A_{>i}$, this implies that $(u,v) \in B^R$ by (D4), a contradiction.

Finally, suppose that $\{u,u'\} \cap X \neq \emptyset$, say $u \in X$. Since $(u,v) \notin B^R$, by (D1) we infer that $u \in R(v)$. Then, by (D3), $vv' \notin E(H)$ and $uu' \in E(G)$ together imply that $(u',v') \in B^R$, a contradiction.

This finishes the proof that ϕ is a solution to (G, rev). To see that $\mathsf{rev}^R(\phi) = \mathsf{rev}(\phi)$ note that ϕ, being an optimum solution to $(G^{x,y}, \mathsf{rev}^R)$, does not use any assignments with negative revenues in rev^R, while $\mathsf{rev}(u,v) = \mathsf{rev}^R(u,v)$ for all (u,v) satisfying $\mathsf{rev}^R(u,v) \geq 0$. ■

CLAIM 4.8. *If ϕ is a solution to (G, rev) that is compatible with a guess $R \in \mathcal{R}^{x,y}$, then ϕ is also a solution to $(G^{x,y}, \mathsf{rev}^R)$ and $\mathsf{rev}^R(\phi) = \mathsf{rev}(\phi)$.*

Proof of Claim. As ϕ is a solution to (G, rev), it is a partial H-coloring of G. Since $G^{x,y}$ is a subgraph of G with $V(G^{x,y}) = V(G)$, ϕ is also a partial H-coloring of $G^{x,y}$. Hence ϕ is a solution to $(G^{x,y}, \mathsf{rev}^R)$.

To prove that $\mathsf{rev}^R(\phi) = \mathsf{rev}(\phi)$ it suffices to show that $(u, \phi(u)) \notin B^R$ for every $u \in \mathsf{dom}\,\phi$, since functions rev^R and rev differ only on the pairs from B^R. Suppose otherwise, and consider cases depending on the reason for including $(u, \phi(u))$ in B^R. Denote $v \coloneqq \phi(u)$.

First, suppose $u \in X$ and $u \notin R(v)$. By (C2) we have $u \notin R(v) \cap X = \phi^{-1}(v) \cap X \ni u$, a contradiction.

Second, suppose $u \in R(v')$ for some $v' \neq v$. By (C1) we have $v = \phi(u) = v'$, again a contradiction.

Third, suppose that u has a neighbor u' in G such that $u' \in R(v')$ for some $v' \in V(H)$ satisfying $vv' \notin E(H)$. By (C1), we have $u' \in \mathsf{dom}\,\phi$ and $\phi(u') = v'$. But then $\phi(u)\phi(u') = vv' \notin E(H)$ even though $uu' \in E(G)$, a contradiction with the assumption that ϕ is a partial H-coloring of G.

Fourth, suppose that $u \in A_i \setminus R(v)$ for some $i \in \{1,2,3\}$ and there exists $u' \in A_{>i}$ such that $uu' \in E(G)$ and $N_G(u') \cap R(v) \cap A_i = \emptyset$. Observe that since $u \in A_i \cap \phi^{-1}(v)$ and $uu' \in E(G)$, by (C3) u' has a neighbor in $R(v) \cap A_i$ in the graph G. This contradicts the supposition that $N_G(u') \cap R(v) \cap A_i = \emptyset$.

As in all the cases we have obtained a contradiction, this concludes the proof of the claim. ■

We now relate the optimum solution to the instance (G, rev) to optima for instances constructed for different $(x,y) \in T$. For $(x,y) \in T$, consider a a set of instances

$$\Lambda^{x,y} \coloneqq \{\, (G^{x,y}, \mathsf{rev}^R) \;:\; R \in \mathcal{R}^{x,y} \,\},$$

and let $\Lambda \coloneqq \bigcup_{(x,y) \in T} \Lambda^{x,y}$. Note that

$$|\Lambda| \leq |T| \cdot n^{\mathcal{O}(\mathsf{Ramsey}(s,t))} \leq (|V(H)| \cdot n) \cdot n^{\mathcal{O}(\mathsf{Ramsey}(s,t))} \leq n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}.$$

We then have the following.

CLAIM 4.9. *We have* $\mathrm{OPT}(G, \mathsf{rev}) = \max_{(G', \mathsf{rev}') \in \Lambda} \mathrm{OPT}(G', \mathsf{rev}')$. *Moreover, for every $(G', \mathsf{rev}') \in \Lambda$ for which the maximum is reached, every optimum solution ϕ to (G', rev') is also an optimum solution to (G, rev) with $\mathsf{rev}(\phi) = \mathsf{rev}'(\phi)$.*

Proof of Claim. By Claim 4.7, we have that

$$\mathrm{OPT}(G, \mathsf{rev}) \geq \max_{(G', \mathsf{rev}') \in \Lambda} \mathrm{OPT}(G', \mathsf{rev}'). \tag{4.1}$$

On the other hand, suppose $\phi^\star$ is an optimum solution to (G, rev). Since $T \neq \emptyset$ by assumption, hence there exists some $(x,y) \in T$ such that $\phi^\star(x) = y$. By

Claim 4.5, there exists a guess $R \in \mathcal{R}^{x,y}$ such that $\phi^\star$ is compatible with R; note that $(G^{x,y}, \mathsf{rev}^R) \in \Lambda$. By Claim 4.8, $\phi^\star$ is also a solution to the instance $(G^{x,y}, \mathsf{rev}^R)$ and $\mathsf{rev}^R(\phi^\star) = \mathsf{rev}(\phi^\star)$. By (4.1) we conclude that $\phi^\star$ is an optimum solution to $(G^{x,y}, \mathsf{rev}^R)$ and $\mathrm{OPT}(G, \mathsf{rev}) = \mathrm{OPT}(G^{x,y}, \mathsf{rev}^R)$. In particular, $\mathrm{OPT}(G, \mathsf{rev}) = \max_{(G', \mathsf{rev}') \in \Lambda} \mathrm{OPT}(G', \mathsf{rev}')$. Finally, Claim 4.7 now implies that every optimum solution to $(G^{x,y}, \mathsf{rev}^R)$ is also an optimum solution to (G, rev). ■

Claim 4.9 asserts that the instance (G, rev) is suitably equivalent to the set of instances Λ. It now remains to partition each instance from Λ into two independent subinstances (G_1, rev_1) and (G_2, rev_2) with properties required in (B1) and (B2), so that the final set Π can be obtained by applying this operation to every instance in Λ.

Consider any instance from Λ, say instance $(G^{x,y}, \mathsf{rev}^R)$ constructed for some $(x, y) \in T$ and $R \in \mathcal{R}^{x,y}$. We adopt the notation from the construction of $G^{x,y}$ and $\mathcal{R}^{x,y}$, and define

$$G_1^{x,y} \coloneqq G^{x,y}[A_1] \qquad \text{and} \qquad G_2^{x,y} \coloneqq G^{x,y}[A_2 \cup A_3 \cup A_4 \cup X].$$

The properties of $G_1^{x,y}$ and $G_2^{x,y}$ required in (B1) and (B2) are asserted by the following claim.

Claim 4.10. *The graphs $G_1^{x,y}$ and $G_2^{x,y}$ are $\{P_6, L_s, S_t\}$-free. Moreover, for every connected graph F on at least two vertices, if $G_1^{x,y}$ contains an induced F, then G contains an induced $F^\bullet$, and if $G_2^{x,y}$ contains an induced F, then G contains an induced $F^{\bullet\!-\!\circ}$.*

Proof of Claim. Note that $G_1^{x,y}$ is an induced subgraph of G. Moreover, $G_2^{x,y}$ is a disjoint union of $G[A_2]$, $G[A_3]$, and $G[A_4]$, plus x_1, x_2, x_3 are included in $G_2^{x,y}$ as isolated vertices, so every connected component of $G_2^{x,y}$ is an induced subgraph of G. As G is $\{P_6, L_s, S_t\}$-free by assumption, it follows that both $G_1^{x,y}$ and $G_2^{x,y}$ are $\{P_6, L_s, S_t\}$-free. The second part of the statement follows directly from Claim 4.2 and the observation that every induced F in $G_2^{x,y}$ has to be contained either in $G[A_2]$, or in $G[A_3]$, or in $G[A_4]$. ■

Now, construct an instance $(G_1^{x,y}, \mathsf{rev}_1^R)$ of Max Partial H'-Coloring, where $H' = H - y$, and an instance $(G_2^{x,y}, \mathsf{rev}_2^R)$ of Max Partial H-Coloring as follows: rev_1^R is defined as the restriction of rev^R to the set $V(G_1^{x,y}) \times V(H')$, and rev_2^R is defined as the restriction of rev^R to the set $V(G_2^{x,y}) \times V(H)$. Note that by Claim 4.6 and the construction of rev^R, we have $\mathsf{rev}^R(u, y) = -1$ for all $u \in V(G_1^{x,y})$, so no optimum solution to $(G^{x,y}, \mathsf{rev}^R)$ can assign y to any $u \in V(G_1^{x,y})$. Since in $G^{x,y}$ there are no edges between $V(G_1^{x,y})$ and $V(G_2^{x,y})$, we immediately obtain the following.

Claim 4.11. $\mathrm{OPT}(G^{x,y}, \mathsf{rev}^R) = \mathrm{OPT}(G_1^{x,y}, \mathsf{rev}_1^R) + \mathrm{OPT}(G_2^{x,y}, \mathsf{rev}_2^R)$. *Moreover, for any optimum solutions ϕ_1 and ϕ_2 to $(G_1^{x,y}, \mathsf{rev}_1^R)$ and $(G_2^{x,y}, \mathsf{rev}_2^R)$, respectively, the function $\phi \coloneqq \phi_1 \cup \phi_2$ is an optimum solution to $(G^{x,y}, \mathsf{rev}^R)$.*

Finally, we define the set Π to comprise of all the pairs $((G_1^{x,y}, \mathsf{rev}_1^R), (G_2^{x,y}, \mathsf{rev}_2^R))$ constructed from all $(G^{x,y}, \mathsf{rev}^R) \in \Lambda$ as described above. Now, assertion (B3) follows directly from Claim 4.9 and Claim 4.11, while assertions (B1) and (B2) are verified by Claim 4.10.

It remains to argue the algorithmic aspects. There are at most $|V(H)| \cdot n = \mathcal{O}(n)$ pairs $(x, y) \in T$ to consider, and for each of them we can enumerate the set of guesses $\mathcal{R}^{x,y}$ in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$. It is clear that for each guess $R \in \mathcal{R}^{x,y}$, the instances $(G_1^{x,y}, \mathsf{rev}_1^R)$ and $(G_2^{x,y}, \mathsf{rev}_2^R)$ can be constructed in polynomial time. Hence the total running time of $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ follows. This completes the proof of Lemma 4.1.

A simplified variant. In the next section we will rely only on the following simplified variant of Lemma 4.1. We provide it for the convenience of further use.

LEMMA 4.12. *Let H be a fixed irreflexive pattern graph. Suppose we are given integers s, t and an instance (G, rev) of* MAX PARTIAL H-COLORING *such that G is connected and $\{P_6, L_s, S_t\}$-free. Denoting $n := |V(G)|$, one can in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ construct a subgraph G' of G with $V(G') = V(G)$ and a set Π consisting of at most $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ revenue functions with domain $V(G) \times V(H)$ such that the following conditions hold:*

(C1) The graph G' is $\{P_6, L_s, S_t\}$-free. Moreover, if G is $F^\bullet$-free for some connected graph F on at least two vertices, then G' is F-free.

(C2) We have $\mathrm{OPT}(G, \mathsf{rev}) = \max_{\mathsf{rev}' \in \Pi} \mathrm{OPT}(G', \mathsf{rev}')$. Moreover, for any $\mathsf{rev}' \in \Pi$ for which the maximum is reached, every optimum solution ϕ to (G', rev') is also an optimum solution to (G, rev) with $\mathsf{rev}(\phi) = \mathsf{rev}'(\phi)$.

Proof. The proof is a simplified version of the proof of Lemma 4.1, hence we only highlight the differences.

First, we do not iterate through all the pair $(x, y) \in T$: we perform only one construction of a subgraph G' and a set of guesses $\mathcal{R}$, which is analogous to the construction of $G^{x,y}$ and $\mathcal{R}^{x,y}$ for a single pair (x, y) from the proof of Lemma 4.1. For X we just take any set of three vertices such that $N[X]$ is a monitor in G, and we enumerate X as $\{x_1, x_2, x_3\}$ in any way. The remainder of the construction proceeds as before, resulting in a family of guesses $\mathcal{R}$ of size $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ and a subgraph G' of G (the graph $G^{x,y}$ from the proof of Lemma 4.1). Here, in the definition of a guess we omit the condition that $\phi(x) = y$; this does not affect the asymptotic bound on the number of guesses. A subset of the reasoning presented in the proofs of Claim 4.2 and Claim 4.10 shows that G' is $\{P_6, L_s, S_t\}$-free and, moreover, for every connected graph F on at least two vertices, if G' contains an induced F, then G contains an induced $F^\bullet$. Note that since we are interested only in finding an induced $F^\bullet$ instead of $F^{\bullet\!-\!\circ}$, we do not need edges between vertices of X for this. This verifies assertion (C1). If we now define $\Pi := \{\mathsf{rev}^R \colon R \in \mathcal{R}\}$, then the same reasoning as in Claim 4.9 verifies assertion (C2). Note here that Claim 4.7 and Claim 4.8 are still valid verbatim after replacing $G^{x,y}$ by G' and $\mathcal{R}^{x,y}$ by $\mathcal{R}$. ∎

5. Exhaustive branching. In this section we give the first set of corollaries that can be derived from Lemma 4.1. The idea is to apply this tool exhaustively, until the considered instance becomes trivial. The main point is that with each application the clique number of the graph drops, hence we naturally obtain an upper bound of the form of $n^{f(\omega(G))}$ for the total size of the recursion tree, hence also on the running time. This leads to results (R3) and (R4) promised in Section 1. In fact, we will only rely on the simplified variant of Lemma 4.1, that is, Lemma 4.12.

The following statement captures the idea of exhaustive applying Lemma 4.12 in a recursive scheme. For the convenience of further use, we formulate the following statement so that s and t are given on input.

THEOREM 5.1. *Let H be a fixed irreflexive pattern graph. There exists an algorithm that given $s, t \in \mathbb{N}$ and an instance (G, rev) of* MAX PARTIAL H-COLORING *where G is $\{P_6, L_s, S_t\}$-free, solves this instance in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t) \cdot \omega(G))}$.*

Proof. If G is not connected, then for every connected component C of G we apply the algorithm recursively to $(C, \mathsf{rev}|_{V(C)})$. If ϕ_C is the computed optimum solution to this instance, we may output $\phi := \bigcup_C \phi_C$. It is clear that ϕ constructed in this way is an optimum solution to the instance (G, rev).

Assume then that G is connected. If G consists of only one vertex, say u, then we may simply output $\phi \coloneqq \{(u, v)\}$ where v maximizes $\mathsf{rev}(u, v)$, or $\phi \coloneqq \emptyset$ if $\mathsf{rev}(\cdot)$ has no positive value in its range. Hence, assume that G has at least two vertices, in particular $\omega(G) \geq 2$. We now apply Lemma 4.12 to G. Thus, in time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ we obtain a subgraph G' of G with $V(G) = V(G')$ and a suitable set of revenue functions Π satisfying $|\Pi| \leq n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$. Recall here that G' is $\{P_6, L_s, S_t\}$-free. Moreover, if we set $F = K_{\omega(G)}$ then G is $F^\bullet$-free, so Lemma 4.12 implies that G' is F-free. This means that $\omega(G') < \omega(G)$.

Next, for every $\mathsf{rev}' \in \Pi$ we recursively solve the instance (G', rev'). Lemma 4.12 then implies that if among the obtained optimum solutions to instances (G', rev') we pick the one with the largest revenue, then this solution is also an optimum solution to (G, rev) that can be output by the algorithm.

We are left with analyzing the running time. Recall that every time we recurse into subproblems constructed using Lemma 4.12, the clique number of the currently considered graph drops by at least one. Since recursing on a disconnected graph yields connected graphs in subproblems, we conclude that the total depth of the recursion tree is bounded by $2 \cdot \omega(G)$. In every recursion step we branch into $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$ subproblems, hence the total number of nodes in the recursion tree is bounded by $\left(n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}\right)^{2\cdot\omega(G)} = n^{\mathcal{O}(\mathsf{Ramsey}(s,t)\cdot\omega(G))}$. The internal computation in each subproblem take time $n^{\mathcal{O}(\mathsf{Ramsey}(s,t))}$, hence the total running time is indeed $n^{\mathcal{O}(\mathsf{Ramsey}(s,t)\cdot\omega(G))}$. ◻

Note that since both L_3 and S_2 contain P_5 as an induced subgraph, every P_5-free graph is $\{P_6, L_3, S_2\}$-free. Hence, from Theorem 5.1 we may immediately conclude the following statement, where the setting of P_5-free graphs is covered by the case $s = 3$ and $t = 2$.

COROLLARY 5.2. *For any fixed $s, t \in \mathbb{N}$ and irreflexive pattern graph H,* MAX PARTIAL H-COLORING *can be solved in $\{P_6, L_s, S_t\}$-free graphs in time $n^{\mathcal{O}(\omega(G))}$. This in particular applies to P_5-free graphs.*

Next, we observe that the statement of Theorem 5.1 can be also used for non-constant s to obtain an algorithm for the case when the graph L_s is not excluded.

COROLLARY 5.3. *For any fixed $t \in \mathbb{N}$ and irreflexive pattern graph H,* MAX PARTIAL H-COLORING *can be solved in $\{P_6, S_t\}$-free graphs in time $n^{\mathcal{O}(\omega(G)^t)}$.*

Proof. Observe that since the graph L_s contains a clique of size s, every graph G is actually $L_{\omega(G)+1}$-free. Therefore, we may apply the algorithm of Theorem 5.1 for $s \coloneqq \omega(G) + 1$. Note here that $\omega(G)$ can be computed in time $n^{\omega(G)+\mathcal{O}(1)}$ by verifying whether G has cliques of size $1, 2, 3, \ldots$ up to the point when the check yields a negative answer. Since for $s = \omega(G) + 1$ and fixed t we have

$$\mathsf{Ramsey}(s, t) = \binom{s + t - 2}{t - 1} \leq \mathcal{O}(\omega(G)^{t-1}),$$

the obtained running time is $n^{\mathcal{O}(\mathsf{Ramsey}(s,t)\cdot\omega(G))} \leq n^{\mathcal{O}(\omega(G)^t)}$. ◻

Let us note that an algorithm with running time $n^{\mathcal{O}(\omega(G)^\alpha)}$, for some constant α, can be used within a simple branching strategy to obtain a subexponential-time algorithm.

LEMMA 5.4. *Let H be a fixed irreflexive graph and suppose* MAX PARTIAL H-COLORING *can be solved in time $n^{\mathcal{O}(\omega(G)^\alpha)}$ on $\mathcal{F}$-free graphs, for some family of* *graphs $\mathcal{F}$ and some constant $\alpha \geq 1$. Then* Max Partial H-Coloring *can be solved in time $n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$ on $\mathcal{F}$-free graphs.*

Proof. Let (G, rev) be the input instance, where G has n vertices. We define threshold $\tau := \left\lfloor n^{\frac{1}{\alpha+1}} \right\rfloor$. We assume that $\tau > |V(H)|$, for otherwise the instance has constant size and can be solved in constant time.

The algorithm first checks whether G contains a clique on τ vertices. This can be done in time $n^{\tau+\mathcal{O}(1)} \leq n^{\mathcal{O}(n^{1/(\alpha+1)})}$ by verifying all subsets of τ vertices in G. If there is no such clique then $\omega(G) < \tau$, so we can solve the problem using the assumed algorithm in time $n^{\mathcal{O}(\omega(G)^\alpha)} \leq n^{\mathcal{O}(\tau^\alpha)} \leq n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$. Hence, suppose that we have found a clique K on τ vertices.

Observe that since H is irreflexive, in any partial H-coloring ϕ of G only at most $|V(H)|$ vertices of K can be colored, that is, belong to $\mathsf{dom}\,\phi$. We recurse into $\binom{\tau}{\leq |V(H)|} \leq n^{|V(H)|}$ subproblems: in each subproblem we fix a different subset $A \subseteq K$ with $|A| \leq |V(H)|$ and recurse on the graph $G_A := G - (K \setminus A)$ with revenue function $\mathsf{rev}_A := \mathsf{rev}|_{V(G_A)}$. Note here that G_A is $\mathcal{F}$-free. From the above discussion it is clear that $\mathrm{OPT}(G, \mathsf{rev}) = \max_{A \subseteq K, |A| \leq |V(H)|} \mathrm{OPT}(G_A, \mathsf{rev}_A)$. Therefore, the algorithm may return the solution with the highest revenue among those obtained in recursive calls.

As for the running time, observe that in every recursive call, the algorithm either solves the problem in time $n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$, or recurses into $n^{|V(H)|} = n^{\mathcal{O}(1)}$ subcalls, where in each subcall the vertex count is decremented by at least $\left\lfloor n^{\frac{1}{\alpha+1}} \right\rfloor - |V(H)|$. It follows that the depth of the recursion is bounded by $\mathcal{O}(n^{\alpha/(\alpha+1)})$, hence the total number of nodes in the recursion tree is at most $n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$. Since the time used for each node is bounded by $n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$, the total running time of $n^{\mathcal{O}(n^{\alpha/(\alpha+1)})}$ follows. ∎

By combining Corollary 5.2 and Corollary 5.3 with Lemma 5.4 we conclude the following.

Corollary 5.5. *For any fixed $s, t \in \mathbb{N}$ and irreflexive pattern graph H,* Max Partial H-Coloring *can be solved in $\{P_6, L_s, S_t\}$-free graphs in time $n^{\mathcal{O}(\sqrt{n})}$. This in particular applies to P_5-free graphs.*

Corollary 5.6. *For any fixed $t \in \mathbb{N}$ and irreflexive pattern graph H,* Max Partial H-Coloring *can be solved in $\{P_6, S_t\}$-free graphs in time $n^{\mathcal{O}\left(n^{t/(t+1)}\right)}$.*

6. Excluding a threshold graph. We now present the next result promised in Section 1, namely result (R1): the problem is polynomial-time solvable on $\{P_5, F\}$-free graphs whenever F is a threshold graph. For this, we observe that a *constant* number of applications of Lemma 4.1 reduces the input instance to instances that can be solved trivially. Thus, the whole recursion tree has polynomial size, resulting in a polynomial-time algorithm. Note that here we use the full, non-simplified variant of Lemma 4.1.

We have the following statement.

Theorem 6.1. *Fix $s, t \in \mathbb{N}$. Suppose F is a connected graph on at least two vertices such that for every fixed irreflexive pattern graph H, the* Max Partial H-Coloring *problem can be solved in polynomial time in $\{P_6, L_s, S_t, F\}$-free graphs. Then for every fixed irreflexive pattern graph H, the* Max Partial H-Coloring *problem can be solved in polynomial time in $\{P_6, L_s, S_t, F^{\bullet\!-\!\circ}\}$-free graphs.*

Proof. We proceed by induction on $|V(H)|$, hence we assume that for all proper induced subgraphs H' of H, MAX PARTIAL H'-COLORING admits a polynomial-time algorithm on $\{P_6, L_s, S_t, F^{\bullet\!-\!\circ}\}$-free graphs. Here, the base case is given by H being the empty graph; then the empty function is the only solution.

Let (G, rev) be an input instance (G, rev) of MAX PARTIAL H-COLORING, where G is $\{P_6, L_s, S_t, F^{\bullet\!-\!\circ}\}$-free. We may assume that G is connected, as otherwise we may apply the algorithm to each connected component of G separately, and output the union of the obtained solutions. Further, if the range of rev contains only non-positive numbers, then the empty function is an optimum solution to (G, rev); hence assume otherwise.

We may now apply Lemma 4.1 to (G, rev) to construct a suitable list of instances Π. Note that since s and t are considered fixed, Π has polynomial size and can be computed in polynomial time. Consider any pair $((G_1, \mathsf{rev}_1), (G_2, \mathsf{rev}_2)) \in \Pi$. On one hand, (G_1, rev_1) is a $\{P_6, L_s, S_t, F\}$-free instance of MAX PARTIAL H'-COLORING where H' is some proper induced subgraph of H, so we can apply an algorithm from the inductive assumption to solve it in polynomial time. On the other hand, as G is $F^{\bullet\!-\!\circ}$-free, from Lemma 4.1 it follows that G_2 is $\{P_6, L_s, S_t, F\}$-free. Therefore, by assumption, the instance (G_2, rev_2) can be solved in in polynomial time.

Finally, by Lemma 4.1, to obtain an optimum solution to (G, rev) it suffices to take the highest-revenue solution obtained as the union of optimum solutions to instances in some pair from Π. As the size of Π is polynomial and each of the instances involved in Π can be solved in polynomial time, we can output an optimum solution to (G, rev) in polynomial time. ∎

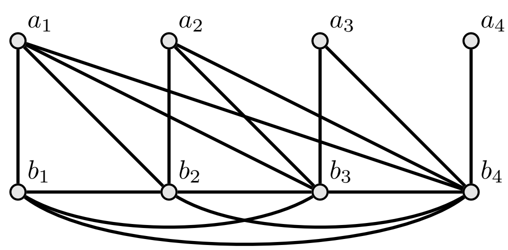

FIG. 6. *The graph Q_4.*

Let us define a graph Q_k as follows, see Figure 6. The vertex set consists of two disjoint sets $A \coloneqq \{a_1, \ldots, a_k\}$ and $B \coloneqq \{b_1, \ldots, b_k\}$. The set A is independent in Q_k, while B is turned into a clique. The adjacency between A and B is defined as follows: for $i, j \in \{1, \ldots, k\}$, we make a_i and b_j adjacent if and only if $i \le j$. Note that Q_k is a threshold graph.

We now use Theorem 6.1 to prove the following.

COROLLARY 6.2. *For every fixed $k, s, t \in \mathbb{N}$ and irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem can be solved in polynomial time in $\{P_6, L_s, S_t, Q_k\}$-free graphs. This in particular applies to $\{P_5, Q_k\}$-free graphs.*

Proof. It suffices to observe that $Q_{k+1} = (Q_k)^{\bullet\!-\!\circ}$ and apply induction on k. Note that the base case for $k = 1$ holds trivially, because $Q_1 = K_2$, so in this case we consider the class of edgeless graphs. As before, the last point of the statement follows by taking $s = 3$ and $t = 2$ and noting that both L_3 and S_2 contain an induced P_5. ∎

It is straightforward to observe that for every threshold graph F there exists

$k \in \mathbb{N}$ such that F is an induced subgraph of H_k. Therefore, from Corollary 6.2 we can derive the following.

COROLLARY 6.3. *For every fixed threshold graph F and irreflexive pattern graph H,* MAX PARTIAL H-COLORING *can be solved in polynomial time in $\{P_5, F\}$-free graphs.*

We now note that in Corollary 6.2 we started the induction with $Q_1 = K_2$, however we could also apply the reasoning starting from any other graph F for which we know that MAX PARTIAL H-COLORING can be solved in polynomial time in $\{P_6, L_s, S_t, F\}$-free graphs. One such example is $F = P_4$, for which we can derive polynomial-time solvability using a different argument.

LEMMA 6.4. *For every fixed irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem in P_4-free graphs can be solved in polynomial time.*

Proof. It is well-known that P_4-free graphs are exactly *cographs*, which in particular have cliquewidth at most 2 (and a suitable clique expression can be computed in polynomial time). Therefore, we can solve MAX PARTIAL H-COLORING in cographs in polynomial time using the meta-theorem of Courcelle, Makowsky, and Rotics [18] for MSO_1-expressible optimization problems on graphs of bounded cliquewidth. This is because for a fixed H, it is straightforward to express MAX PARTIAL H-COLORING as such a problem. Alternatively, one can write an explicit dynamic programming algorithm, which is standard. ◻

By applying the same reasoning as in Corollary 6.2, but starting the induction with P_4, we conclude:

COROLLARY 6.5. *Suppose F is a graph obtained from P_4 by a repeated application of the $(\cdot)^{\bullet\!-\!\circ}$ operator. Then for every fixed irreflexive pattern graph H,* MAX PARTIAL H-COLORING *can be solved in polynomial time in $\{P_5, F\}$-free graphs.*

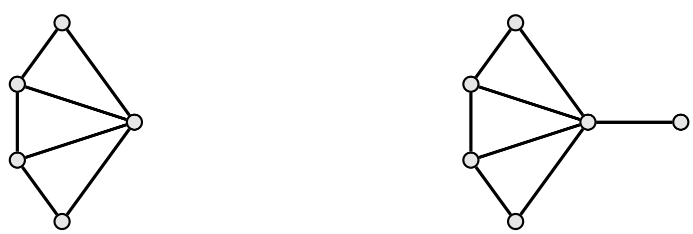

FIG. 7. *The gem and the graph $(P_4)^{\bullet\!-\!\circ}$.*

We note here that $(P_4)^{\bullet\!-\!\circ}$ is the graph obtained from the *gem* graph by adding a degree-one vertex to the center of the gem; see Figure 7. It turns out that $\{P_5, \text{gem}\}$-free graphs have bounded cliquewidth [6], hence the polynomial-time solvability of MAX PARTIAL H-COLORING on these graphs follows from the same argument as that used for P_4-free graphs in Lemma 6.4. However, this argument does not apply to any of the cases captured by Corollary 6.5. Indeed, as shown in [8, Theorem 25(v)], $\{F_1, F_2\}$-free graphs have unbounded cliquewidth (and even mim-width) whenever both F_1 and F_2 contain an independent set of size 3, and both P_5 and $(P_4)^{\bullet\!-\!\circ}$ enjoy this property. Note that this argument can be also applied to infer that $\{P_5, \text{bull}\}$-free graphs have unbounded cliquewidth and mim-width, which is the setting that we explore in the next section.

7. Excluding a bull. In this section we prove result (R2) promised in Section 1. The technique is similar in spirit to that used in Section 6. Namely, we apply Lemma 4.1 twice to reduce the problem to the case of P_4-free graphs, which can be handled using Lemma 6.4. However, these applications are interleaved with a reduction to the case when the input graph is *prime*: it does not contain any non-trivial *module* (equivalently, *homogeneous set*). This allows us to use some combinatorial results about the structure of prime bull-free graphs [16, 14].

7.1. Reduction to prime graphs. In order to present the reduction to the case of prime graphs it will be convenient to work with a *multicoloring* generalization of the problem. In this setting, we allow mapping vertices of the input graph G to nonempty subset of vertices of H, rather than to single vertices of H.

Multicoloring variant. For a graph H, by $\mathsf{Pow}^\star(H)$ we denote the set of all nonempty subsets of $V(H)$. Let H be an irreflexive pattern graph and G be a graph. A *partial H-multicoloring* is a partial function $\phi\colon V(G) \rightharpoonup \mathsf{Pow}^\star(H)$ that satisfies the following condition: for every edge $uu' \in E(G)$ such that $u, u' \in \mathsf{dom}\,\phi$, the sets $\phi(u), \phi(u') \subseteq V(H)$ are disjoint and complete to each other in H; that is, $vv' \in E(G)$ for all $v \in \phi(u)$ and $v' \in \phi(u')$. We correspondingly redefine the measurement of revenue. A *revenue function* is a function $\mathsf{rev}\colon V(G) \times \mathsf{Pow}^\star(H) \to \mathbb{R}$ and the revenue of a partial H-multicoloring ϕ is defined as

$$\mathsf{rev}(\phi) \coloneqq \sum_{u \in \mathsf{dom}\,\phi} \mathsf{rev}(u, \phi(u)).$$

The Max Partial H-Multicoloring problem is then defined as follows.

Max Partial H-Multicoloring
Input: Graph G and a revenue function $\mathsf{rev}\colon V(G) \times \mathsf{Pow}^\star(H) \to \mathbb{R}$
Output: A partial H-multicoloring ϕ of G that maximizes $\mathsf{rev}(\phi)$

Clearly, Max Partial H-Multicoloring generalizes Max Partial H-Coloring, as given an instance (G, rev) of Max Partial H-Coloring, we can turn it into an equivalent instance (G, rev') of Max Partial H-Multicoloring by defining rev' as follows: for $u \in V(G)$ and $Z \subseteq V(H)$, we set

$$\mathsf{rev}'(u, Z) \coloneqq \begin{cases} \mathsf{rev}(u, v) & \text{if } Z = \{v\} \text{ for some } v \in V(H); \\ -1 & \text{otherwise.} \end{cases}$$

However, there is actually also a reduction in the other direction. For an irreflexive pattern graph H, we define another pattern graph $\widehat{H}$ as follows: $V(\widehat{H}) = \mathsf{Pow}^\star(H)$ and we make $X, Y \in \mathsf{Pow}^\star(H)$ adjacent in $\widehat{H}$ if and only if X and Y are disjoint and complete to each other in H. Note that $\widehat{H}$ is again irreflexive and since we consider H fixed, $\widehat{H}$ is a constant-sized graph. Then it is easy to see that the set of instances of Max Partial H-Multicoloring is exactly equal to the set of instances of Max Partial $\widehat{H}$-Coloring, and the definitions of solutions and their revenues coincide. Thus, we may solve instances of Max Partial H-Multicoloring by applying algorithms for Max Partial $\widehat{H}$-Coloring to them. Let us remark that expressing Max Partial H-Multicoloring as Max Partial $\widehat{H}$-Coloring is similar to expressing k-tuple coloring (or fractional coloring) as homomorphisms to Kneser graphs, see e.g. [31, Section 6.2].

Modular decompositions. We are mostly interested in MAX PARTIAL H-MULTICOLORING because in this general setting, it is easy to reduce the problem once we find a non-trivial *module* (or *homogeneous set*) in an instance. For clarity, we choose to present this approach by performing dynamic programming on a modular decomposition of the input graph, hence we need a few definitions. The following standard facts about modular decompositions can be found for instance in the survey of Habib and Paul [30].

A *module* (or a *homogeneous set*) in a graph G is a subset of vertices B such that every vertex $u \notin B$ is either complete of anti-complete to B. A module B is *proper* if $2 \leq |B| < |V(G)|$. A graph G is called *prime* if it does not have any proper modules.

A module B in a graph G is *strong* if for any other module B', we have either $B \subseteq B'$, or $B \supseteq B'$, or $B \cap B' = \emptyset$. It is known that if among proper strong modules in a graph G we choose the (inclusion-wise) maximal ones, then they form a partition of the vertex set of G, called the *modular partition* $\mathsf{Mod}(G)$. The *quotient graph* $\mathsf{Quo}(G)$ is the graph with $\mathsf{Mod}(G)$ as the vertex set where two maximal proper strong modules $B, B' \in \mathsf{Mod}(G)$ are adjacent if they are complete to each other in G, and non-adjacent if they are anti-complete to each other in G. It is known that for every graph G, the quotient graph $\mathsf{Quo}(G)$ is either edgeless, or complete, or prime. Note that the quotient graph $\mathsf{Quo}(G)$ is always an induced subgraph of G: selecting one vertex from each element of $\mathsf{Mod}(G)$ yields a subset of vertices that induces $\mathsf{Quo}(G)$ in G.

The *modular decomposition* of a graph is a tree $\mathcal{T}$ whose nodes are modules of G, which is constructed by applying modular partitions recursively. First, created a root node $V(G)$. Then, as long as the current tree has a leaf B with $|B| \geq 2$, attach the elements of $\mathsf{Mod}(G[B])$ as children of B. Thus, the leaves of $\mathcal{T}$ exactly contain all single-vertex modules of G; hence $\mathcal{T}$ has n leaves and at most $2n-1$ nodes in total. It is known that the set of nodes of the modular decomposition of G exactly comprises of all the strong modules in G. Moreover, given G, the modular decomposition of G can be computed in linear time [19, 36].

Dynamic programming on modular decomposition. The following lemma shows that given a graph G, MAX PARTIAL H-MULTICOLORING in G can be solved by solving the problem for each element of $\mathsf{Mod}(G)$, and combining the results by solving the problem on $\mathsf{Quo}(G)$. Here, H is an irreflexive pattern graph that we fix from this point on.

LEMMA 7.1. *Let* (G, rev) *be an instance of* MAX PARTIAL H-MULTICOLORING, *where H is irreflexive. For* $B \in \mathsf{Mod}(G)$ *and* $W \in \mathsf{Pow}^\star(H)$*, define* $\mathsf{rev}_{B,W} \colon B \times \mathsf{Pow}^\star(H) \to \mathbb{R}$ *as follows: for* $u \in B$ *and* $Z \in \mathsf{Pow}^\star(H)$*, set*

$$\mathsf{rev}_{B,W}(u, Z) \coloneqq \begin{cases} \mathsf{rev}(u, Z) & \text{if } Z \subseteq W; \\ -1 & \text{otherwise.} \end{cases}$$

Further, define $\mathsf{rev}' \colon \mathsf{Mod}(G) \times \mathsf{Pow}^\star(H) \to \mathbb{R}$ *as follows: for* $B \in \mathsf{Mod}(G)$ *and* $W \in \mathsf{Pow}^\star(H)$*, set*

$$\mathsf{rev}'(B, W) \coloneqq \mathrm{OPT}(G[B], \mathsf{rev}_{B,W}).$$

Then $\mathrm{OPT}(G, \mathsf{rev}) = \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}')$*. Moreover, for every optimum solution* ϕ' *to* $(\mathsf{Quo}(G), \mathsf{rev}')$ *and optimum solutions* ϕ_B *to respective instances* $(G[B], \mathsf{rev}_{B,\phi'(B)})$*, for* $B \in \mathsf{Mod}(G) \cap \mathsf{dom}\, \phi'$*, the function*

$$\phi \coloneqq \bigcup_{B \in \mathsf{Mod}(G) \cap \mathsf{dom}\, \phi'} \phi_B$$

is an optimum solution to (G, rev).

Proof. We first argue that $\mathrm{OPT}(G, \mathsf{rev}) \le \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}')$. Take an optimum solution ϕ to (G, rev). For every $B \in \mathsf{Mod}(G)$, let

$$\phi'(B) \coloneqq \bigcup_{u \in B \cap \mathsf{dom}\,\phi} \phi(u),$$

unless the right hand side is equal to $\emptyset$, in which case we do not include B in the domain of ϕ'. Observe that ϕ' defined in this manner is a solution to the instance $(\mathsf{Quo}(G), \mathsf{rev}')$. Indeed, if for some $BB' \in E(\mathsf{Quo}(G))$ we did not have that $\phi'(B)$ and $\phi'(B')$ are disjoint and complete to each other in H, then there would exist $u \in B$ and $u' \in B'$ such that $\phi(u)$ and $\phi(u')$ are not disjoint and complete to each other in H, contradicting the assumption that ϕ is a solution to (G, rev).

Note that for each $B \in \mathsf{dom}\,\phi'$, $\phi|_B$ is a solution to the instance $(G[B], \mathsf{rev}_{B,\phi'(B)})$. Observe that

$$\mathrm{OPT}(G, \mathsf{rev}) = \mathsf{rev}(\phi) = \sum_{B \in \mathsf{dom}\,\phi'} \mathsf{rev}_{B,\phi'(B)}(\phi|_B) \le \sum_{B \in \mathsf{dom}\,\phi'} \mathrm{OPT}(G[B], \mathsf{rev}_{B,\phi'(B)}),$$

where the second equality follows from the fact that rev and $\mathsf{rev}_{B,\phi'(B)}$ agree on all pairs $(u, \phi(u))$ for $u \in B \cap \mathsf{dom}\,\phi$. On the other hand, since ϕ' is a solution to $(\mathsf{Quo}(G), \mathsf{rev}')$, we have

$$\sum_{B \in \mathsf{dom}\,\phi'} \mathrm{OPT}(G[B], \mathsf{rev}_{B,\phi'(B)}) = \sum_{B \in \mathsf{dom}\,\phi'} \mathsf{rev}'(B, \phi'(B))$$
$$= \mathsf{rev}'(\phi') \le \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}').$$

This proves that $\mathrm{OPT}(G, \mathsf{rev}) \le \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}')$.

Next, we argue that $\mathrm{OPT}(G, \mathsf{rev}) \ge \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}')$ and that the last assertion from the lemma statement holds. Let ϕ' be an optimum solution to the instance $(\mathsf{Quo}(G), \mathsf{rev}')$. Further, for each $B \in \mathsf{dom}\,\phi'$, let ϕ_B be any optimum solution to the instance $(G[B], \mathsf{rev}_{B,\phi'(B)})$. Consider

$$\phi \coloneqq \bigcup_{B \in \mathsf{dom}\,\phi'} \phi_B$$

We verify that ϕ is a solution to (G, rev). The only non-trivial check is that for any $B, B' \in \mathsf{dom}\,\phi'$ with $BB' \in E(\mathsf{Quo}(G))$, $u \in \mathsf{dom}\,\phi_B$, and $u' \in \mathsf{dom}\,\phi_{B'}$, we have that $\phi(u)$ and $\phi(u')$ are disjoint and complete to each other in H. However, ϕ_B, as an optimal solution to $(G[B], \mathsf{rev}_{B,\phi'(B)})$, does not use any assignments with negative revenues, which implies that $\phi(u) = \phi_B(u) \subseteq \phi'(B)$. Similarly, we have $\phi(u') = \phi_{B'}(u') \subseteq \phi'(B')$. Since $\phi'(B)$ and $\phi'(B')$ are disjoint and complete to each other, due to the assumption that ϕ' is a solution to $(\mathsf{Quo}(G), \mathsf{rev}')$, the same can be also claimed about $\phi(u)$ and $\phi(u')$.

Finally, observe that

$$\mathsf{rev}(\phi) = \sum_{B \in \mathsf{dom}\,\phi'} \mathsf{rev}_{B,\phi'(B)}(\phi_B)$$
$$= \sum_{B \in \mathsf{dom}\,\phi'} \mathrm{OPT}(G[B], \mathsf{rev}_{B,\phi'(B)}) = \mathsf{rev}'(\phi') = \mathrm{OPT}(\mathsf{Quo}(G), \phi'),$$

where the first equality follows from the fact that rev and $\mathsf{rev}_{B,\phi'(B)}$ agree on all assignments used by ϕ, for all $B \in \mathsf{dom}\,\phi'$. This proves that

$$\mathrm{OPT}(G, \mathsf{rev}) \geq \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}').$$

Combining this inequality with the with the reverse one proved before, we conclude that $\mathrm{OPT}(G, \mathsf{rev}) = \mathrm{OPT}(\mathsf{Quo}(G), \mathsf{rev}')$ and ϕ is an optimum solution to (G, rev). ◻

Lemma 7.1 enables us to perform dynamic programming on a modular decomposition, provided the problem can be solved efficiently on prime graphs from the considered graph class. This leads to the following statement.

LEMMA 7.2. *Let H be a fixed irreflexive pattern graph. Let $\mathcal{F}$ be a set of graphs such that* MAX PARTIAL H-MULTICOLORING *can be solved in time $T(n)$ on prime $\mathcal{F}$-free graphs. Then* MAX PARTIAL H-MULTICOLORING *can be solved in time $n^{\mathcal{O}(1)} \cdot T(n)$ on $\mathcal{F}$-free graphs.*

Proof. First, in linear time we compute the modular decomposition $\mathcal{T}$ of G. Then, for every strong module B of G and every $W \in \mathsf{Pow}^{\star}(H)$, we will compute an optimum solution $\phi_{B,W}$ to the instance $(G[B], \mathsf{rev}_{B,W})$, where the revenue function $\mathsf{rev}_{B,W}$ is defined as in Lemma 7.1. At the end, we may return $\phi_{V(G),V(H)}$ as the optimum solution to (G, rev).

The computation of solutions $\phi_{B,W}$ is organized in a bottom-up manner over the decomposition $\mathcal{T}$. Thus, whenever we compute solution $\phi_{B,W}$ for a strong module B and $W \in \mathsf{Pow}^{\star}(H)$, we may assume that the solutions $\phi_{B',W'}$ for all $B' \in \mathsf{Mod}(G[B])$ and $W' \in \mathsf{Pow}^{\star}(H)$ have already been computed.

When B is a leaf of $\mathcal{T}$, say $B = \{u\}$ for some $u \in V(G)$, then for every $W \in \mathsf{Pow}^{\star}(W(H))$ we may simply output $\phi_{B,W} \coloneqq \{(u, Z)\}$ where Z maximizes $\mathsf{rev}_{B,W}(u, Z)$, or $\phi_{B,W} \coloneqq \emptyset$ if $\mathsf{rev}_{B,W}$ has no positive values in its range.

Now suppose B is a non-leaf node of $\mathcal{T}$ and $W \in \mathsf{Pow}^{\star}(W(H))$. Construct an instance $(\mathsf{Quo}(G[B]), \mathsf{rev}')$ similarly as in the statement of Lemma 7.1: for $B' \in \mathsf{Mod}(G[B])$ and $Z \in \mathsf{Pow}^{\star}(H)$, we put

$$\mathsf{rev}'(B, W) \coloneqq \mathrm{OPT}(G[B'], \mathsf{rev}_{B',W\cap Z}).$$

Note here that the values $\mathrm{OPT}(G[B'], \mathsf{rev}_{B',W\cap Z})$ have already been computed, as they are equal to $\mathsf{rev}_{B',W\cap Z}(\phi_{B',W\cap Z})$. From Lemma 7.1 applied to the instance $(G[B], \mathsf{rev}_{W,B})$ it follows that if ϕ' is an optimum solution to $(\mathsf{Quo}(G[B]), \mathsf{rev}')$, then the union of solutions $\phi_{B',\phi'(B')}$ over all $B' \in \mathsf{dom}\,\phi'$ is an optimum solution to $(G[B], \mathsf{rev}_{B,W})$. Therefore, it remains to solve the instance $(\mathsf{Quo}(G[B]), \mathsf{rev}')$. We make a case distinction depending on whether $\mathsf{Quo}(G[B])$ is edgeless, complete, or prime.

It is very easy to argue that MAX PARTIAL H-MULTICOLORING can be solved in polynomial time both in edgeless graphs and in complete graphs. For instance, one can equivalently see the instance as an instance of MAX PARTIAL $\widehat{H}$-COLORING, and apply the algorithm for P_4-free graphs given by Lemma 6.4.

On the other hand, if $\mathsf{Quo}(G[B])$ is prime, then by assumption we can solve the instance $(\mathsf{Quo}(G[B]), \mathsf{rev}')$ in time $T(n)$. Recall here that $\mathsf{Quo}(G[B])$ is an induced subgraph of $G[B]$, hence it is also $\mathcal{F}$-free.

This concludes the description of the algorithm. As for the running time, observe that since H is considered fixed, the computation for each node of the decomposition take time $n^{\mathcal{O}(1)} \cdot T(n)$. Since $\mathcal{T}$ has at most $2n - 1$ nodes, the total running time of $n^{\mathcal{O}(1)} \cdot T(n)$ follows. ◻

We can now conclude the following statement. Note that it speaks only about the standard variant of the MAX PARTIAL H-COLORING problem.

THEOREM 7.3. *Let $\mathcal{F}$ be a set of graphs such that for every fixed irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem can be solved in polynomial time in prime $\mathcal{F}$-free graphs. Then for every fixed irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem can be solved in polynomial time in $\mathcal{F}$-free graphs.*

Proof. As instances of MAX PARTIAL H-MULTICOLORING can be equivalently regarded as instances of MAX PARTIAL $\widehat{H}$-COLORING, we conclude that for every fixed H, MAX PARTIAL H-MULTICOLORING is polynomial-time solvable in prime $\mathcal{F}$-free graphs — just apply the algorithm for MAX PARTIAL $\widehat{H}$-COLORING. By Lemma 7.2 we infer that for every fixed H, MAX PARTIAL H-MULTICOLORING is polynomial-time solvable in $\mathcal{F}$-free graphs. As MAX PARTIAL H-MULTICOLORING generalizes MAX PARTIAL H-COLORING, this algorithm can be used to solve MAX PARTIAL H-COLORING in $\mathcal{F}$-free graphs in polynomial time. ∎

7.2. Algorithms for bull-free classes. We now move to our algorithmic results for subclasses of bull-free graphs. For this, we need to recall some definitions and results.

For graphs F and G, we say that G contains an *induced F with a center and an anti-center* if there exists $A \subseteq V(G)$ such that $G[A]$ is isomorphic to F, and moreover there are vertices $x, y \notin A$ such that x is complete to A and y is anti-complete to A. Observe that if a graph G contains an induced $F^{\bullet\!-\!\circ}$, then G contains an induced F with a center and an anti-center. We will use the following.

THEOREM 7.4 ([16]). *Let G be a $\{bull, C_5\}$-free graph. If G contains an induced P_4 with a center and an anti-center, then G is not prime.*

THEOREM 7.5 ([14]). *Let G be a bull-free graph. If G contains an induced C_5 with a center and an anti-center, then G is not prime.*

We now combine Lemma 4.12, Theorem 7.3, and Theorem 7.4 to show the following.

LEMMA 7.6. *For every fixed $t \in \mathbb{N}$ and irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem in $\{P_6, C_5, S_t, bull\}$-free graphs can be solved in polynomial time.*

Proof. As in the proof of Theorem 6.1, we proceed by induction on $|V(H)|$. Hence, we assume that for all proper induced subgraphs H' of H, MAX PARTIAL H'-COLORING can be solved in polynomial-time on $\{P_6, C_5, S_t, \text{bull}\}$-free graphs. By Theorem 7.3, it suffices to give a polynomial-time algorithm for MAX PARTIAL H-COLORING working on prime $\{P_6, C_5, S_t, \text{bull}\}$-free graphs. By Theorem 7.4, such graphs do not contain any induced P_4 with a center and an anti-center, so in particular they do not contain any induced $(P_4)^{\bullet\!-\!\circ}$.

Consider then an input instance (G, rev) of MAX PARTIAL H-COLORING, where G is $\{P_6, C_5, S_t, \text{bull}\}$-free and prime, hence also connected. If the range of rev consists only of non-positive numbers, then the empty function is an optimum solution to (G, rev), hence assume otherwise. Note that L_3 contains an induced bull, hence we may apply Lemma 4.12 for $s = 3$ to compute a suitable set Π of pairs of instances. This takes polynomial time due to t being considered a constant.

Consider any pair $((G_1, \mathsf{rev}_1), (G_2, \mathsf{rev}_2)) \in \Pi$. On one hand, (G_1, rev_1) is an instance of MAX PARTIAL H'-COLORING for some proper induced subgraph H' of H, hence we can apply an algorithm from the inductive assumption to solve it in polynomial time. On the other hand, note that the graph G_2 is P_4-free, for if it had an induced P_4, then by Lemma 4.12 we would find an induced $(P_4)^{\bullet\!-\!\circ}$ in G, a contradiction to G being prime by Theorem 7.4. Hence, we can solve the instance (G_2, rev_2) in polynomial time using the algorithm of Lemma 6.4.

Finally, Lemma 4.12 implies that to obtain an optimum solution to (G, rev), it suffices to take the highest-revenue solution obtained as the union of optimum solutions to instances in some pair from Π. Since the size of Π is polynomial and each of the instances involved in Π can be solved in polynomial time, we can output an optimum solution to (G, rev) in polynomial time as well. ◻

Finally, it remains to combine Lemma 7.6 with Lemma 4.12 again to derive the main result of this section.

THEOREM 7.7. *For every fixed $t \in \mathbb{N}$ and irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem in $\{P_6, S_t, bull\}$-free graphs can be solved in polynomial time.*

Proof. We follow exactly the same strategy as in the proof of Lemma 7.6. The differences are that:

- Instead of using Theorem 7.4, we apply Theorem 7.5 to argue that the graph G_2 is C_5-free.
- Instead of using Lemma 6.4 to solve P_4-free instances, we apply Lemma 7.6 to solve $\{P_6, C_5, S_t, \text{bull}\}$-free instances.

The straightforward application of these modifications is left to the reader. ◻

Finally, since $S_2 = P_5$, from Theorem 7.7 we immediately conclude the following.

COROLLARY 7.8. *For every fixed irreflexive pattern graph H, the* MAX PARTIAL H-COLORING *problem in $\{P_5, bull\}$-free graphs can be solved in polynomial time.*

8. Hardness for patterns with loops. Recall that the assumption that H is irreflexive is crucial in our approach in Lemma 4.1. However, while H-COLORING becomes trivial if H has loops, this is no longer the case for generalizations of the problem, including LIST H-COLORING and MAX PARTIAL H-COLORING. See e.g. [22, 29, 39].

Here, LIST H-COLORING is the list variant of the H-COLORING problem: an instance of LIST H-COLORING is a pair (G, L), where G is a graph and $L\colon V(G) \to 2^{V(H)}$ assigns a *list* to every vertex. We ask whether G admits an H-coloring ϕ that respects lists L, i.e., $\phi(v) \in L(v)$ for every $v \in V(G)$.

Note that that LIST H-COLORING is a special case of MAX PARTIAL H-COLORING: for any instance (G, L) of LIST H-COLORING, define the revenue function $\mathsf{rev}\colon V(G) \times V(H) \to \mathbb{R}$ as follows:

$$\mathsf{rev}(v,u) = \begin{cases} -1 & \text{if } u \notin L(v); \\ 1 & \text{if } u \in L(v). \end{cases}$$

It is straightforward to observe that solving the instance (G, L) of LIST H-COLORING is equivalent to deciding if the instance (G, rev) of MAX PARTIAL H-COLORING has a solution of revenue at least (in fact, equal to) $|V(G)|$. Thus any positive result for MAX PARTIAL H-COLORING can be applied to LIST H-COLORING, while any hardness result for LIST H-COLORING carries over to MAX PARTIAL H-COLORING.

Let us point out that if we only aim for solving LIST H-COLORING, a simple adaptation of the algorithm of Hoàng et al. [32] shows that the problem is polynomial-time solvable in P_5-free graphs, provided H has no loops. In this section we show that there is little hope to extend this positive result to graphs H with loops allowed.

A graph G is a *split graph* if $V(G)$ can be partitioned into a clique and an independent set (that we call the *independent part*). Is is well-known that split graphs are precisely $\{P_5, C_4, 2P_2\}$-free graphs.

Let H_0 be the graph on the vertex set $\bigcup_{i\in\{1,2,3\}}\{a_i, b_i, c_i, d_i\}$ (see Figure 8). The edge set $E(H_0)$ consists of the edges:

- all edges with both endpoints in $\bigcup_{i\in\{1,2,3\}}\{a_i, b_i\}$ (including loops),
- all edges with both endpoints in $\bigcup_{i\in\{1,2,3\}}\{c_i, d_i\}$ (including loops),
- for each $i \in \{1, 2, 3\}$, the edges a_ic_i and b_ic_i,
- for each $i \in \{1, 2, 3\}$ and $j \in \{1, 2, 3\} \setminus \{i\}$, the edges d_ia_i and d_ib_j.

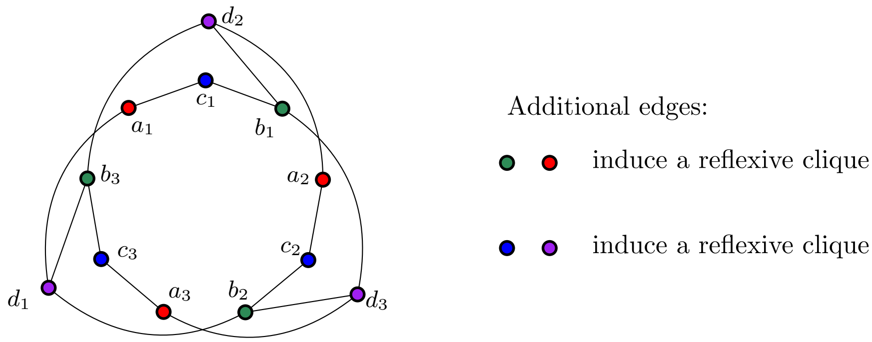

Fig. 8. *The graph H_0 used in Theorem 8.1.*

Theorem 8.1. *The* List H_0-Coloring *problem (and thus* Max Partial H_0-Coloring*) is* NP*-hard and, under the ETH, cannot be solved in time $2^{o(n)}$:*

(a) in split graphs, even if each vertex of the independent part is of degree 2; and

(b) in complements of bipartite graphs (in particular, in $\{P_5$, bull$\}$-free graphs).

Proof. We partition the vertices of H_0 into sets A, B, C, D, where $A \coloneqq \{a_1, a_2, a_3\}$ and the remaining sets are defined analogously.

We reduce from 3-Coloring, which is NP-complete and cannot be solved in time $2^{o(n+m)}$ unless the ETH fails, where n and m respectively denote the number of vertices and of edges [20]. Let G be an instance of 3-Coloring with n vertices and m edges. Let $V(G) = \{v_1, v_2, \ldots, v_n\}$ and let $[n] \coloneqq \{1, \ldots, n\}$.

First, let us build a split graph G' with lists L, which admits an H_0-coloring respecting L if and only if G is 3-colorable. For each $i \in [n]$, we add to G' two vertices x_i and y_i. Let $X \coloneqq \{x_i : i \in [n]\}$ and $Y \coloneqq \{y_i : i \in [n]\}$. We make $X \cup Y$ into a clique in G'. We set $L(x_i) \coloneqq \{a_1, a_2, a_3\}$ and $L(y_i) \coloneqq \{b_1, b_2, b_3\}$ for every $i \in [n]$.

The intended meaning of an H_0-coloring of G' is that for any $i \in [n]$ and $j \in \{1, 2, 3\}$, coloring x_i with color a_j and y_i with color b_j corresponds to coloring v_i with color j. So we need to ensure the following two properties:

(P1) for every $i \in [n]$ and $j \in \{1, 2, 3\}$, the vertex x_i is colored a_j if and only if the vertex y_i is colored b_j,

(P2) for every edge v_iv_j of G, the vertices x_i and x_j get different colors (and, by Item P1, so do y_i and y_j).

In order to ensure property Item P1, for each $i \in [n]$ we introduce a vertex w_i, adjacent to x_i and y_i, whose list is $\{c_1, c_2, c_3\}$. By W we denote the set $\{w_i : i \in [n]\}$. To ensure property Item P2, for each edge v_iv_j of G, where $i < j$, we introduce a vertex $z_{i,j}$

adjacent to x_i and y_j. The list of $z_{i,j}$ consists of $\{d_1, d_2, d_3\}$. By Z we denote the set $\{z_{i,j} : v_iv_j \in E(G) \text{ and } i < j\}$.

It is straightforward to verify that the definition of the neighborhoods of vertices c_i, d_i in H_0 forces Item P1 and Item P2, which implies that G is 3-colorable if and only if G' admits an H_0-coloring that respects lists L. The number of vertices of G' is

$$|X| + |Y| + |W| + |Z| = n + n + n + m = \mathcal{O}(n + m).$$

Hence, if the obtained instance of the List H_0-Coloring problem could be solved in time $2^{o(|V(G')|)}$, then this would imply the existence of a $2^{o(n+m)}$-time algorithm for 3-Coloring, a contradiction with the ETH. Furthermore, $X \cup Y$ is a clique, $W \cup Z$ is independent, and every vertex from $W \cup Z$ has degree 2. Thus the statement (a) of the theorem holds.

We observe that the set $\{L(v) : v \in W \cup Z\} = C \cup D$ forms a reflexive clique in H_0. Thus we can turn the set $W \cup Z$ into a clique, obtaining an equivalent instance (G'', L) of List H_0-Coloring. As the vertex set of G'' can be partitioned into two cliques, G'' is the complement of a bipartite graph, so the statement (b) of the theorem holds as well. □

9. Open problems. The following question, which originally motivated our work, still remains unresolved.

Question 9.1. Is there a polynomial-time algorithm for Odd Cycle Transversal in P_5-free graphs?

Note that our work stops short of giving a positive answer to this question: we give an algorithm with running time $n^{\mathcal{O}(\omega(G))}$, a subexponential-time algorithm, and polynomial time algorithms for the cases when either a threshold graphs or a bull is additionally forbidden. Therefore, we are hopeful that the answer to the question is indeed positive.

One aspect of our work that we find particularly interesting is the possibility of treating the clique number $\omega(G)$ as a progress measure for an algorithm, which enables bounding the recursion depth in terms of $\omega(G)$. This approach naturally leads to algorithms with running time of the form $n^{f(\omega(G))}$ for some function f, that is, polynomial-time for every fixed clique number. By Lemma 5.4, having a polynomial function f in the above implies the existence of a subexponential-time algorithm, at least in the setting of Max Partial H-Coloring for irreflexive H. However, looking for algorithms with time complexity $n^{f(\omega(G))}$ seems to be another relaxation of the goal of polynomial-time solvability, somewhat orthogonal to subexponential-time algorithms [4, 7, 27] or approximation schemes [13]. Note that our work and the recent work of Brettell et al. [9] actually show two different methods of obtaining such algorithms: using direct recursion, or via dynamic programming on branch decompositions of bounded mim-width. It would be interesting to investigate this direction in the context of Maximum Independent Set in P_t-free graphs. A concrete question would be the following.

Question 9.2. Is there a polynomial-time algorithm for Maximum Independent Set in $\{P_t, K_t\}$-free graphs, for every fixed t?

In all our algorithms, we state the time complexity assuming that the pattern graph H is fixed. This means that the constants hidden in the $\mathcal{O}(\cdot)$ notation in the exponent may — and do — depend on the size of H. In the language of parameterized complexity, this means that we give XP algorithms for the parameterization by the size of H. It is natural to ask whether this state of art can be improved to the existence of FPT algorithms, that is, with running time $f(H) \cdot n^c$ for some computable function f and universal constant c, independent of H. This is not known even for the case of k-Coloring P_5-free graphs, so let us re-iterate the old question of Hoàng et al. [32] (see also [11, Problem 4.1]).

Question 9.3. Is there an FPT algorithm for k-Coloring in P_5-free graphs parameterized by k?

While the above question seems hard, it is conceivable that FPT results could be derived in some more restricted settings, for instance for $2P_2$-free graphs of $\{P_5, \text{bull}\}$-free graphs.

Finally, recall that List H-Coloring in P_5-free graphs is polynomial-time solvable for irreflexive H, but might become NP-hard when loops on H are allowed (see Theorem 8.1). We believe that it would be interesting to obtain a full complexity dichotomy.

Question 9.4. For what pattern graphs H (with possible loops) is List H-Coloring polynomial-time solvable in P_5-free graphs?

We think that solving all problems listed above might require obtaining new structural results, and thus may lead to better understanding of the structure of P_5-free graphs.

Acknowledgments. We acknowledge the welcoming and productive atmosphere at Dagstuhl Seminar 19271 "Graph Colouring: from Structure to Algorithms", where this work has been initiated.

REFERENCES

[1] T. Abrishami, M. Chudnovsky, M. Pilipczuk, P. Rzążewski, and P. D. Seymour, *Induced subgraphs of bounded treewidth and the container method*, in Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021, D. Marx, ed., SIAM, 2021, pp. 1948–1964, https://doi.org/10.1137/1.9781611976465.116, https://doi.org/10.1137/1.9781611976465.116.

[2] V. E. Alekseev, *The effect of local constraints on the complexity of determination of the graph independence number*, Combinatorial-algebraic methods in applied mathematics, (1982), pp. 3–13. (in Russian).

[3] V. E. Alekseev, *Polynomial algorithm for finding the largest independent sets in graphs without forks*, Discret. Appl. Math., 135 (2004), pp. 3–16, https://doi.org/10.1016/S0166-218X(02)00290-1, https://doi.org/10.1016/S0166-218X(02)00290-1.

[4] G. Bacsó, D. Lokshtanov, D. Marx, M. Pilipczuk, Z. Tuza, and E. J. van Leeuwen, *Subexponential-time algorithms for Maximum Independent Set in P_t-free and broom-free graphs*, Algorithmica, 81 (2019), pp. 421–438, https://doi.org/10.1007/s00453-018-0479-5, https://doi.org/10.1007/s00453-018-0479-5.

[5] F. Bonomo, M. Chudnovsky, P. Maceli, O. Schaudt, M. Stein, and M. Zhong, *Three-coloring and list three-coloring of graphs without induced paths on seven vertices*, Combinatorica, 38 (2018), pp. 779–801, https://doi.org/10.1007/s00493-017-3553-8, https://doi.org/10.1007/s00493-017-3553-8.

[6] A. Brandstädt, H. Le, and R. Mosca, *Chordal co-gem-free and (P_5, gem)-free graphs have bounded clique-width*, Discret. Appl. Math., 145 (2005), pp. 232–241, https://doi.org/10.1016/j.dam.2004.01.014, https://doi.org/10.1016/j.dam.2004.01.014.

[7] C. Brause, *A subexponential-time algorithm for the Maximum Independent Set problem in P_t-free graphs*, Discret. Appl. Math., 231 (2017), pp. 113–118, https://doi.org/10.1016/j.dam.2016.06.016, https://doi.org/10.1016/j.dam.2016.06.016.

[8] N. Brettell, J. Horsfield, A. Munaro, G. Paesani, and D. Paulusma, *Bounding the mim-width of hereditary graph classes*, in 15th International Symposium on Parameterized and Exact Computation, IPEC 2020, December 14-18, 2020, Hong Kong, China (Virtual

Conference), Y. Cao and M. Pilipczuk, eds., vol. 180 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, pp. 6:1–6:18, https://doi.org/10.4230/LIPIcs.IPEC.2020.6, https://doi.org/10.4230/LIPIcs.IPEC.2020.6.

[9] N. Brettell, J. Horsfield, and D. Paulusma, *Colouring $sP_1 + P_5$-free graphs: a mim-width perspective*, CoRR, abs/2004.05022 (2020), https://arxiv.org/abs/2004.05022, https://arxiv.org/abs/2004.05022.

[10] M. Chudnovsky, J. King, M. Pilipczuk, P. Rzążewski, and S. Spirkl, *Finding large H-colorable subgraphs in hereditary graph classes*, in 28th Annual European Symposium on Algorithms, ESA 2020, September 7-9, 2020, Pisa, Italy (Virtual Conference), F. Grandoni, G. Herman, and P. Sanders, eds., vol. 173 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, pp. 35:1–35:17, https://doi.org/10.4230/LIPIcs.ESA.2020.35, https://doi.org/10.4230/LIPIcs.ESA.2020.35.

[11] M. Chudnovsky, D. Paulusma, and O. Schaudt, *Graph colouring: from structure to algorithms (Dagstuhl Seminar 19271)*, Dagstuhl Reports, 9 (2019), pp. 125–142, https://doi.org/10.4230/DagRep.9.6.125, https://doi.org/10.4230/DagRep.9.6.125.

[12] M. Chudnovsky, M. Pilipczuk, M. Pilipczuk, and S. Thomassé, *Quasi-polynomial time approximation schemes for the maximum weight independent set problem in H-free graphs*, CoRR, abs/1907.04585 (2019), http://arxiv.org/abs/1907.04585, https://arxiv.org/abs/1907.04585.

[13] M. Chudnovsky, M. Pilipczuk, M. Pilipczuk, and S. Thomassé, *Quasi-polynomial time approximation schemes for the Maximum Weight Independent Set problem in H-free graphs*, in Proceedings of the 31st ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, SIAM, 2020, pp. 2260–2278, https://doi.org/10.1137/1.9781611975994.139, https://doi.org/10.1137/1.9781611975994.139.

[14] M. Chudnovsky and S. Safra, *The Erdős-Hajnal conjecture for bull-free graphs*, J. Comb. Theory, Ser. B, 98 (2008), pp. 1301–1310, https://doi.org/10.1016/j.jctb.2008.02.005, https://doi.org/10.1016/j.jctb.2008.02.005.

[15] M. Chudnovsky, O. Schaudt, S. Spirkl, M. Stein, and M. Zhong, *Approximately coloring graphs without long induced paths*, Algorithmica, 81 (2019), pp. 3186–3199, https://doi.org/10.1007/s00453-019-00577-6, https://doi.org/10.1007/s00453-019-00577-6.

[16] M. Chudnovsky and V. Sivaraman, *Odd holes in bull-free graphs*, SIAM J. Discrete Math., 32 (2018), pp. 951–955, https://doi.org/10.1137/17M1131301, https://doi.org/10.1137/17M1131301.

[17] M. Chudnovsky, S. Spirkl, and M. Zhong, *Four-coloring P_6-free graphs*, in Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, SIAM, 2019, pp. 1239–1256, https://doi.org/10.1137/1.9781611975482.76, https://doi.org/10.1137/1.9781611975482.76.

[18] B. Courcelle, J. A. Makowsky, and U. Rotics, *Linear time solvable optimization problems on graphs of bounded clique-width*, Theory Comput. Syst., 33 (2000), pp. 125–150, https://doi.org/10.1007/s002249910009, https://doi.org/10.1007/s002249910009.

[19] A. Cournier and M. Habib, *A new linear algorithm for modular decomposition*, in Proceedings of the 19th International Colloquium on Trees in Algebra and Programming, CAAP 1994, vol. 787 of Lecture Notes in Computer Science, Springer, 1994, pp. 68–84, https://doi.org/10.1007/BFb0017474, https://doi.org/10.1007/BFb0017474.

[20] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh, *Parameterized Algorithms*, Springer, 2015, https://doi.org/10.1007/978-3-319-21275-3, https://doi.org/10.1007/978-3-319-21275-3.

[21] K. K. Dabrowski, C. Feghali, M. Johnson, G. Paesani, D. Paulusma, and P. Rzazewski, *On cycle transversals and their connected variants in the absence of a small linear forest*, Algorithmica, 82 (2020), pp. 2841–2866, https://doi.org/10.1007/s00453-020-00706-6, https://doi.org/10.1007/s00453-020-00706-6.

[22] T. Feder, P. Hell, and J. Huang, *Bi-arc graphs and the complexity of list homomorphisms*, Journal of Graph Theory, 42 (2003), pp. 61–80, https://doi.org/10.1002/jgt.10073, https://doi.org/10.1002/jgt.10073.

[23] F. V. Fomin, I. Todinca, and Y. Villanger, *Large induced subgraphs via triangulations and CMSO*, SIAM J. Comput., 44 (2015), pp. 54–87, https://doi.org/10.1137/140964801, https://doi.org/10.1137/140964801.

[24] P. Gartland and D. Lokshtanov, *Independent set on P_k-free graphs in quasi-polynomial time*, in 61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020, IEEE, 2020, pp. 613–624, https://doi.org/10.1109/FOCS46700.2020.00063, https://doi.org/10.1109/FOCS46700.2020.00063.

[25] P. Gartland, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, and P. Rzążewski, *Finding*

large induced sparse subgraphs in $C_{>t}$ -free graphs in quasipolynomial time, in STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021, S. Khuller and V. V. Williams, eds., ACM, 2021, pp. 330–341, https://doi.org/10.1145/3406325.3451034, https://doi.org/10.1145/3406325.3451034.

[26] P. A. Golovach, D. Paulusma, and J. Song, *Closing complexity gaps for coloring problems on H-free graphs*, Inf. Comput., 237 (2014), pp. 204–214, https://doi.org/10.1016/j.ic.2014.02.004, https://doi.org/10.1016/j.ic.2014.02.004.

[27] C. Groenland, K. Okrasa, P. Rzążewski, A. D. Scott, P. D. Seymour, and S. Spirkl, *H-colouring P_t-free graphs in subexponential time*, Discret. Appl. Math., 267 (2019), pp. 184–189, https://doi.org/10.1016/j.dam.2019.04.010, https://doi.org/10.1016/j.dam.2019.04.010.

[28] A. Grzesik, T. Klimošová, M. Pilipczuk, and M. Pilipczuk, *Polynomial-time algorithm for Maximum Weight Independent Set on P_6-free graphs*, in Proceedings of the 30^{th} Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, SIAM, 2019, pp. 1257–1271, https://doi.org/10.1137/1.9781611975482.77, https://doi.org/10.1137/1.9781611975482.77.

[29] G. Z. Gutin, P. Hell, A. Rafiey, and A. Yeo, *A dichotomy for minimum cost graph homomorphisms*, Eur. J. Comb., 29 (2008), pp. 900–911, https://doi.org/10.1016/j.ejc.2007.11.012, https://doi.org/10.1016/j.ejc.2007.11.012.

[30] M. Habib and C. Paul, *A survey of the algorithmic aspects of modular decomposition*, Comput. Sci. Rev., 4 (2010), pp. 41–59, https://doi.org/10.1016/j.cosrev.2010.01.001, https://doi.org/10.1016/j.cosrev.2010.01.001.

[31] P. Hell and J. Nešetřil, *Graphs and homomorphisms*, vol. 28 of Oxford lecture series in mathematics and its applications, Oxford University Press, 2004.

[32] C. T. Hoàng, M. Kamiński, V. Lozin, J. Sawada, and X. Shu, *Deciding k-colorability of P_5-free graphs in polynomial time*, Algorithmica, 57 (2010), pp. 74–81.

[33] S. Huang, *Improved complexity results on k-coloring P_t-free graphs*, Eur. J. Comb., 51 (2016), pp. 336–346, https://doi.org/10.1016/j.ejc.2015.06.005, https://doi.org/10.1016/j.ejc.2015.06.005.

[34] D. Lokshtanov, M. Vatshelle, and Y. Villanger, *Independent set in P_5-free graphs in polynomial time*, in Proceedings of the 25^{th} Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, SIAM, 2014, pp. 570–581, https://doi.org/10.1137/1.9781611973402.43, https://doi.org/10.1137/1.9781611973402.43.

[35] V. V. Lozin and M. Milanič, *A polynomial algorithm to find an independent set of maximum weight in a fork-free graph*, J. Discrete Algorithms, 6 (2008), pp. 595–604, https://doi.org/10.1016/j.jda.2008.04.001, https://doi.org/10.1016/j.jda.2008.04.001.

[36] R. M. McConnell and J. P. Spinrad, *Linear-time modular decomposition and efficient transitive orientation of comparability graphs*, in Proceedings of the 5^{th} Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 1994, ACM/SIAM, 1994, pp. 536–545.

[37] G. J. Minty, *On maximal independent sets of vertices in claw-free graphs*, J. Comb. Theory, Ser. B, 28 (1980), pp. 284–304, https://doi.org/10.1016/0095-8956(80)90074-X, http://dx.doi.org/10.1016/0095-8956(80)90074-X.

[38] J. Novotná, K. Okrasa, M. Pilipczuk, P. Rzążewski, E. J. van Leeuwen, and B. Walczak, *Subexponential-time algorithms for finding large induced sparse subgraphs*, Algorithmica, (2020), https://doi.org/https://doi.org/10.1007/s00453-020-00745-z, https://doi.org/10.1007/s00453-020-00745-z.

[39] K. Okrasa and P. Rzążewski, *Subexponential algorithms for variants of the homomorphism problem in string graphs*, J. Comput. Syst. Sci., 109 (2020), pp. 126–144, https://doi.org/10.1016/j.jcss.2019.12.004, https://doi.org/10.1016/j.jcss.2019.12.004.

[40] M. Pilipczuk, M. Pilipczuk, and P. Rzążewski, *Quasi-polynomial-time algorithm for independent set inP_t-free graphs via shrinking the space of induced paths*, in 4th Symposium on Simplicity in Algorithms, SOSA 2021, Virtual Conference, January 11-12, 2021, H. V. Le and V. King, eds., SIAM, 2021, pp. 204–209, https://doi.org/10.1137/1.9781611976496.23, https://doi.org/10.1137/1.9781611976496.23.

[41] N. Sbihi, *Algorithme de recherche d'un stable de cardinalité maximum dans un graphe sans étoile*, Discrete Mathematics, 29 (1980), pp. 53–76. (in French).

[42] M. Yannakakis and F. Gavril, *The maximum k-colorable subgraph problem for chordal graphs*, Inf. Process. Lett., 24 (1987), pp. 133–137, https://doi.org/10.1016/0020-0190(87)90107-4, https://doi.org/10.1016/0020-0190(87)90107-4.