Sparse induced subgraphs in P_7-free graphs of bounded clique number

Maria Chudnovsky* Jadwiga Czyżewska† Kacper Kluk‡ Marcin Pilipczuk§
Paweł Rzążewski¶

Abstract

Many natural computational problems, including e.g. Max Weight Independent Set, Feedback Vertex Set, or Vertex Planarization, can be unified under an umbrella of finding the largest sparse induced subgraph that satisfies some property definable in CMSO_2 logic. It is believed that each problem expressible with this formalism can be solved in polynomial time in graphs that exclude a fixed path as an induced subgraph. This belief is supported by the existence of a quasipolynomial-time algorithm by Gartland, Lokshtanov, Pilipczuk, Pilipczuk, and Rzążewski [STOC 2021], and a recent polynomial-time algorithm for P_6-free graphs by Chudnovsky, McCarty, Pilipczuk, Pilipczuk, and Rzążewski [SODA 2024].

In this work we extend polynomial-time tractability of all such problems to P_7-free graphs of bounded clique number.

Acknowledgment

The conference version of this work was presented
at the International Symposium on Algorithms and Computation (ISAAC) 2025.

1 Introduction

When studying computationally hard problems, a natural question to investigate is whether they become tractable when input instances are somehow "well-structured". In particular, in algorithmic graph theory, we often study the complexity of certain (hard in general) problems, when restricted to specific graph classes. Significant attention is given to classes that are *hereditary*, i.e., closed under vertex deletion.

Potential maximal cliques. Arguably, the problem that is best studied in this context is Max Weight Independent Set (MWIS) in which we are given a vertex-weighted graph and we ask for a set of pairwise non-adjacent vertices of maximum possible weight. Investigating the complexity of MWIS in restricted graph classes led to discovering numerous new tools and techniques in algorithmic graph theory [24, 21, 18, 16]. One of such general techniques is the framework of *potential maximal cliques* by Bouchitté and Todinca [7, 8]. Intuitively speaking, a potential maximal clique (or PMC for

---

[*] Princeton University, United States Of America (`mchudnov@math.princeton.edu`). Supported by NSF Grant DMS-2348219 and by AFOSR grant FA9550-22-1-0083.

[†] University of Warsaw, Poland (`j.czyzewska@mimuw.edu.pl`). Supported by Polish National Science Centre SONATA BIS-12 grant number 2022/46/E/ST6/00143.

[‡] University of Warsaw, Poland (`k.kluk@mimuw.edu.pl`). Supported by Polish National Science Centre SONATA BIS-12 grant number 2022/46/E/ST6/00143.

[§] University of Warsaw, Poland (`m.pilipczuk@mimuw.edu.pl`). Supported by Polish National Science Centre SONATA BIS-12 grant number 2022/46/E/ST6/00143.

[¶] Warsaw University of Technology & University of Warsaw, Poland (`pawel.rzazewski@pw.edu.pl`). Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme grant agreement number 948057.

short) in a graph G is a bag of a "reasonable" tree decomposition of G (see Section 2 for a formal definition). The key contribution of Bouchitté and Todinca [7, 8] is showing that:

1. the family $\mathcal{F}$ of PMCs in a graph G can be enumerated in time polynomial in $|V(G)|$ and $|\mathcal{F}|$, and
2. given a family $\mathcal{F}$ containing *all* PMCs of G (and possibly some other sets as well), MWIS can be solved in time polynomial in $|V(G)|$ and $|\mathcal{F}|$ by mimicking natural dynamic programming on an appropriate (but unknown) tree decomposition.

Consequently, MWIS admits a polynomial-time algorithm when restricted to any class with polynomially many PMCs. (Here we say that a class $\mathcal{X}$ of graphs has polynomially many PMCs if there exists a polynomial p, such that the number of PMCs in any n-vertex graph in $\mathcal{X}$ is at most $\mathsf{p}(n)$.)

Later this framework was extended by Fomin, Todinca, and Villanger [19] to the problem of finding a large "sparse" (here meaning: of bounded treewidth) induced subgraph satisfying certain CMSO_2 formula ψ. (CMSO_2 stands for *Counting Monadic Second Order* logic, which is a logic where one can use vertex/edge variables, vertex/edge set variables, quantifications over these variables, and standard propositional operands.) Formally, for a given integer d and a fixed CMSO_2 formula ψ with one free set variable, the $(\mathrm{tw} \leq d, \psi)$-MWIS problem (here 'MWIS' stands for 'max weight induced subgraph') is defined as follows.

$(\mathrm{tw} \leq d, \psi)$-MWIS

Input: A graph G equipped with a weight function $\mathfrak{w} \colon V(G) \to \mathbb{Q}_+$.

Task: Find a pair (Sol, X) such that

- $X \subseteq \mathsf{Sol} \subseteq V(G)$,
- $G[\mathsf{Sol}]$ is of treewidth at most d,
- $G[\mathsf{Sol}] \models \psi(X)$,
- X is of maximum weight subject to the conditions above,

or conclude that no such pair exists.

As shown by Fomin, Todinca, and Villanger [19], every problem expressible in this formalism can be solved in polynomial time on classes of graphs with polynomially many potential maximal cliques.

Note that if $d = 0$ and ψ is a formula satisfied by all sets, then $(\mathrm{tw} \leq d, \psi)$-MWIS is exactly MWIS. It turns out that $(\mathrm{tw} \leq d, \psi)$-MWIS captures several other well known computational problems (see also the discussion in [19, 30]), e.g.;

- Feedback Vertex Set (one of Karp's original 21 NP-complete problems [28]), equivalent by complementation to finding an induced forest of maximum weight,
- Even Cycle Transversal [35, 34, 2], equivalent by complementation to finding an induced graph whose every block is an odd cycle,
- finding a largest weight induced subgraph whose every component is a cycle,
- finding a maximum number of pairwise disjoint and non-adjacent induced cycles.

MWIS in graphs excluding a long induced path. Despite all advantages of the Bouchitté-Todinca framework, its applicability is somehow limited, as there are numerous natural hereditary graph classes that *do not* have polynomially many PMCs. Can we use at least some parts of the framework outside its natural habitat?

A notorious open question in algorithmic graph theory is whether MWIS (and, more generally, $(\mathrm{tw} \leq d, \psi)$-MWIS) can be solved in polynomial time in graphs that exclude a fixed path as an induced subgraph. It is believed that this question has an affirmative answer, which is supported by the existence of a quasipolynomial-time algorithm of Gartland, Lokshtanov, Pilipczuk, Pilipczuk, and Rzążewski [20, 37, 22]. However, if it comes to polynomial-time algorithms, we know much less.

A polynomial-time algorithm for MWIS (as well as for many other problems) in P_4-free graphs (i.e., graphs that exclude a 4-vertex path as an induced subgraph; analogously we define P_t-free graphs for any t), was discovered already in 1980s [14]. In today's terms we would say that these graphs have bounded clique-width and thus polynomial-time algorithms for many natural problems, including $(\mathrm{tw} \leq d, \psi)$-MWIS, follow from a celebrated theorem by Courcelle, Makowsky, and Rotics [15]. However, already the P_5-free case proved to be quite challenging. In 2014, Lokshtanov, Vatshelle, and Villanger [31] showed that Max Weight Independent Set admits a polynomial-time algorithm in P_5-free graphs — by adapting the framework of Bouchitté and Todinca. Let us emphasize that P_5-free graphs might have exponentially many PMCs, so the approach discussed above cannot be applied directly. Instead, Lokshtanov, Vatshelle, and Villanger proved that in polynomial time one can enumerate a family of *some* PMCs, and this family is sufficient to solve MWIS via dynamic programming.

By extending this method (and adding a significant layer of technical complicacy), Grzesik, Klimošová, Pilipczuk, and Pilipczuk [23] managed to show that MWIS is polynomially-solvable in P_6-free graphs.

Some time later, Abrishami, Chudnovsky, Pilipczuk, Rzążewski, and Seymour [1] revisited the P_5-free case and introduced a major twist to the method: they proved that in order to solve MWIS and its generalizations, one does not have to have PMCs exactly, but it is sufficient to enumerate their *containers*: supersets that do not introduce any new vertices of the (unknown) optimum solution. They also proved that containers for *all* PMCs in P_5-free graphs can be enumerated in polynomial time, circumventing the problem of exponentially many PMCs and significantly simplifying the approach of Lokshtanov, Vatshelle, and Villanger [31]. This result yields a polynomial-time algorithm for $(\mathrm{tw} \leq d, \psi)$-MWIS in P_5-free graphs.

By further relaxing the notion of a container (to a *carver*), Chudnovsky, McCarty, Pilipczuk, Pilipczuk, and Rzążewski [13] managed to extend polynomial-time solvability of $(\mathrm{tw} \leq d, \psi)$-MWIS to the class of P_6-free graphs.

Our contribution. In this work we push the boundary of tractability of $(\mathrm{tw} \leq d, \psi)$-MWIS in P_t-free graphs by showing the following result.

Theorem 1.1. *For every fixed integers d, k, and a CMSO$_2$ formula ψ, the $(\mathrm{tw} \leq d, \psi)$-MWIS problem can be solved in polynomial time in P_7-free graphs with clique number at most k.*

Quite interestingly, in P_t-free graphs, the boundedness of treewidth is equivalent to the boundedness of degeneracy and to the boundedness of treedepth [4] (see also Section 2). Thus, Theorem 1.1 yields a polynomial-time algorithm for problems like:

- Vertex Planarization [27, 36], which asks for a largest induced planar subgraph, or
- for constant k, finding a largest set of vertices inducing a subgraph of maximum degree at most k (see, e.g., [26]).

Let us remark that the idea of investigating P_t-free graphs of bounded clique number was considered before. Brettel, Horsfield, Munaro, and Paulusma [11] showed that P_5-free graphs of bounded clique number (actually, a superclass of these graphs) have bounded *mim-width*, which gives a uniform approach to solve many classic problems in this class of graphs [3]. Pilipczuk and Rzążewski [38] showed that P_6-free graphs of bounded clique number have polynomially many PMCs and thus the framework of Bouchitté and Todinca can be applied here directly. However, this is no longer true for P_7-free graphs (even bipartite). On the other hand, Brandstädt and Mosca [9] provided a polynomial-time algorithm for MWIS in P_7-free triangle-free graphs. However, they used a rather ad-hoc approach that works well for MWIS, but gives little hope to generalize it to more complicated instances of $(\mathrm{tw} \le d, \psi)$-MWIS. Our work vastly extends the latter two results.

1.1 Technical overview

As mentioned, in P_t-free graphs the width parameters treewidth, treedepth, and degeneracy are functionally equivalent [4]. Furthermore, as discussed in [13], the property of having treewidth, treedepth, or degeneracy at most d can be expressed as a CMSO_2 formula of size depending on d only. Consequently, if we define problems $(\mathrm{td} \le d, \psi)$-MWIS and $(\deg \le d, \psi)$-MWIS analogously as $(\mathrm{tw} \le d, \psi)$-MWIS, but with respect to treedepth and degeneracy, these three formalisms describe the same family of problems, when restricted to P_t-free graphs.

Following [13], the treedepth formalism is the most handy; we henceforth work with $(\mathrm{td} \le d, \psi)$-MWIS. Furthermore, it is more convenient to work not just with induced subgraphs of bounded treedepth, but with *treedepth-d structures*: induced subgraphs of treedepth at most d with a fixed elimination forest; formal definition can be found in Section 2.

Let $\mathcal{T}$ be a treedepth-d structure in G; think of $\mathcal{T}$ as the sought solution to the $(\mathrm{td} \le d, \psi)$-MWIS problem in question. As proven in [19] (and cast onto the current setting in [13]), there exists a tree decomposition (T, β) of G whose every bag is either contained in $N[v]$ for a leaf v of $\mathcal{T}$, or whose intersection with $\mathcal{T}$ is contained in a single root-to-leaf path of $\mathcal{T}$, excluding the leaf. We call the latter bags *$\mathcal{T}$-avoiding*.

Furthermore, the framework of [19] shows how to solve $(\mathrm{td} \le d, \psi)$-MWIS given a family $\mathcal{F}$ of subsets of $V(G)$ with the promise that all bags of such tree decomposition (T, β) are in $\mathcal{F}$. In [1], it is shown that it suffices for $\mathcal{F}$ to contain only *containers of bounded defect* for bags in (T, β): we require that there exists a universal constant $\widehat{d}$ such that for every $t \in V(T)$ there exists $A \in \mathcal{F}$ such that the bag $\beta(t)$ is contained in A and A contains at most $\widehat{d}$ elements of $\mathcal{T}$.

Note that if v is a leaf of $\mathcal{T}$, then $N[v]$ may contain only ancestors of v among the vertices of $\mathcal{T}$, so $N[v]$ is an excellent container for any bag contained in $N[v]$. Thus, it suffices to provide containers for $\mathcal{T}$-avoiding bags. This is how [1] solved $(\mathrm{td} \le d, \psi)$-MWIS in P_5-free graphs; by generalizing the definition of a container to a carver, the same approach is applied to P_6-free graphs in [13].

In fact, for a treedepth-d structure $\mathcal{T}$, the tree decomposition (T, β) of [19] is constructed as follows: it is shown that there exists a minimal chordal completion F of G (i.e., an inclusion-wise minimal set of edges to add to G to make it chordal) whose addition keeps $\mathcal{T}$ a treedepth-d structure, and (T, β) is any clique tree of $G+F$. A *potential maximal clique* (abbreviated as PMC) is a maximal clique in $G+F$ for *some* minimal chordal completion F of G. Both [1] and [13] actually provide containers/carvers for *every* treedepth-d structure in G and an *arbitrary* choice of a $\mathcal{T}$-avoiding PMC in G. Furthermore, [13] provides an example of a family of P_7-free triangle-free graphs where such a statement is impossible (i.e., any such family of containers/carvers would need to be of exponential size).

However, the algorithmic framework of [19, 1, 13] requires only that the provided family $\mathcal{F}$ contains containers/carvers for all bags of *only one* “good” tree decomposition (T, β), and only for the sought solution $\mathcal{T}$. This should be contrasted with all $\mathcal{T}$-avoiding PMCs, which is in some sense the set of all reasonable $\mathcal{T}$-avoiding bags, and for all treedepth-d structures $\mathcal{T}$. Thus, if we want to tackle P_7-free graphs of bounded clique number, we need to significantly deviate from the path of [1, 13] and either use the power of the choice of (T, β) or the fact that we need to handle only $\mathcal{T}$ being the optimum solution to the problem we are solving.

The assumption of bounded clique number allows us to quickly reduce the case of finding a container for a PMC to finding a container for a minimal separator. For a set $S \subseteq V(G)$, a connected component A of $G - S$ is *full to S* if $N(A) = S$; a set S is a *minimal separator* if it has at least two full components. A classic fact about PMCs [8] is that for every potential maximal clique Ω and for every connected component D of $G - \Omega$, the set $N(D)$ is a minimal separator. By a classic result of Gyárfás, the class of P_t-free graphs is χ-bounded, in particular, if G is P_t-free and $\omega(G) \le k$ (where $\omega(G)$ is the number of vertices in a largest clique in G), then $\chi(G) \le (t-1)^{k-1}$. This, combined with a VC-dimension argument based on the classic result of Ding, Seymour, and Winkler [17], shows the following:

Lemma 1.2. *For every t and k there exists c such that for every P_t-free graph G with $\omega(G) \leq k$ and any PMC Ω of G one of the two following conditions holds:*

1. *there exists $v \in \Omega$ such that $\Omega = N[v]$, or*
2. *there exists a family $\mathcal{D} \subseteq \mathrm{cc}(G - \Omega)$ of size at most c such that $\Omega = \bigcup_{D \in \mathcal{D}} N(D)$.*

That is, except for a simple case $N[v] = \Omega$ for some $v \in \Omega$, the PMC Ω is a union of a constant number of minimal separators. Hence, it suffices to generate a family $\mathcal{F}'$ of containers for $\mathcal{T}$-avoiding minimal separators (which are defined analogously to $\mathcal{T}$-avoiding bags) and take $\mathcal{F}$ to be the family of unions of all tuples of at most c elements of $\mathcal{F}'$.

Let G be a P_7-free graph with $\omega(G) \leq k$. We start as in [1, 13]: let $\mathcal{T}$ be an arbitrary treedepth-d structure in G and let S be a minimal separator in G with full components A and B such that S is $\mathcal{T}$-avoiding. We want to design a polynomial-time algorithm that outputs a family $\mathcal{F}'$ of subsets of $V(G)$ that contains a container for Ω. The algorithm naturally does not know $\mathcal{T}$ nor S; the convenient and natural way of describing the algorithm as performing some nondeterministic guesses about $\mathcal{T}$ and S, with the goal of outputting a container for S in the end. We succeed if the number of subcases coming from the guesses is polynomial in the size of G and the family $\mathcal{F}'$ consists of all containers generated by all possible runs of the algorithm.

We almost succeed with this quest in Section 3. That is, we are able to guess an "almost container" K for S: K contains a constant number of vertices of $\mathcal{T}$ and we identified a set $\mathcal{D}$ of *tricky* connected components of $G - K$ that are contained in $A \cup B \cup S$; all other connected components of $G - K$ are contained in A, in B, or in some other connected component of $G - S$. Every tricky component $D \in \mathcal{D}$ is somewhat simpler: we identify a subset $\emptyset \neq L(D) \subsetneq D$ such that if $\mathcal{C}(D)$ is the family of all connected components of $G[L(D)]$ and of $G[D \setminus L(D)]$, then every $C \in \mathcal{C}(D)$ is a module of $G[D]$. (A set $A \subseteq V(G)$ is a module in G if every vertex $v \in V(G) \setminus A$ is either complete to A or anti-complete to A; more on modules in Section 2.)

Observe now that every connected component $C \in \mathcal{C}(D)$ satisfies $\omega(G[C]) < \omega(G) \leq k$, as $D \setminus C$ contains a vertex complete to C. Hence, we can recurse on C, understanding it fully (more precisely, finding optimum partial solutions; the definition of this "partial" requires a lot of CMSO_2 mumbling).

Furthermore, we observe that the quotient graph $G[D]/\mathcal{C}(D)$ is bipartite. Thus, if we can understand how to solve $(\mathrm{td} \leq d, \psi)$-MWIS on P_7-free *bipartite* graphs, then we should be done: the partial solutions from components $C \in \mathcal{C}(D)$, combined with the understanding of the P_7-free bipartite graph $G[D]/\mathcal{C}(D)$ should give all partial solutions of $G[D]$ for every $D \in \mathcal{D}$. This, in turn, should give an understanding on how $(\mathrm{td} \leq d, \psi)$-MWIS behaves on $K \cup \bigcup \mathcal{D}$. As $S \subseteq K \cup \bigcup \mathcal{D}$, this should suffice to solve $(\mathrm{td} \leq d, \psi)$-MWIS on G using $(K, \mathcal{D}, L)$ as an object that plays the role of the container for S.

Explaining properly all "should"s from the previous paragraph is quite involved and tedious and done in Section 5. In this overview, let us focus on the more interesting case of solving $(\mathrm{td} \leq d, \psi)$-MWIS in P_7-free bipartite graphs. Here, we actually prove the following result in Section 4.

Theorem 1.3. *There exists an algorithm that, given a P_7-free bipartite graph G and an integer $d > 0$, runs in time $n^{2^{2^{\mathcal{O}(d^3)}}}$ and computes a family $\mathcal{C}$ of subsets of $V(G)$ with the following guarantee: for every $\mathsf{Sol} \subseteq V(G)$ such that $G[\mathsf{Sol}]$ is of treedepth at most d, there exists a tree decomposition (T, β) of G such that for every $t \in V(T)$ we have $\beta(t) \in \mathcal{C}$ and $|\beta(t) \cap \mathsf{Sol}| = 2^{2^{\mathcal{O}(d^3)}}$.*

We remark that Theorem 1.3 is not needed if one just wants to solve the MWIS problem, as this problem can be solved in general bipartite graphs using matching or flow techniques. Thus, if one is interested only in finding a maximum-weight independent set, Section 4 can be omitted.

On the other hand, our work identified the bipartite case as an interesting subcase in exploring tractability of $(\mathrm{td} \leq d, \psi)$-MWIS in P_t-free graphs. To state a precise problem for future work, we propose polynomial-time tractability of Feedback Vertex Set in P_t-free bipartite graphs. Meanwhile, Theorem 1.3 and its proof in Section 4 can be of independent interest.

Let us also mention that if one is only interested in solving Feedback Vertex Set in P_7-free bipartite graphs, instead of using our Theorem 1.3, one can follow a different approach. Lozin and Zamaraev [33] provided a deep analysis of the structure of bipartite P_7-free graphs which implies that in particular they have bounded *mim-width* and the corresponding decomposition can be found efficiently. This yields a polynomial-time algorithm for Feedback Vertex Set [3]. However, it it far for clear what other cases $(\mathrm{td} \leq d, \psi)$-MWIS can be solved efficiently in bounded-mim-width graphs.

The proof of Theorem 1.3 starts from the work of Kloks, Liu, and Poon [29], who showed that *chordal bipartite* graphs have polynomial number of PMCs, and thus $(\mathrm{td} \leq d, \psi)$-MWIS problem is solvable in this graph class by the direct application of the PMC framework of Bouchitté and Todinca. (A graph G is *chordal bipartite* if it is bipartite and does not contain induced cycles longer than 4.) A P_7-free bipartite graph is *almost* chordal bipartite: it can contain six-vertex cycles.

We would like to add some edges to the input graph G so that it becomes chordal bipartite. Let $C = c_1 - c_2 - \ldots - c_6 - c_1$ be an induced six-vertex cycle in G; we would like to add the edge c_1c_4 to keep G bipartite but break C. We show that, if one chooses C carefully, one can do it so that G remains P_7-free. However, such an addition may break the sought solution $\mathcal{T}$: if $c_1, c_4 \in V(\mathcal{T})$ but are incomparable in the elimination forest $\mathcal{T}$, then $\mathcal{T}$ is no longer a treedepth-d structure in $G + \{c_1c_4\}$.

We remedy this by a thorough investigation of the structure of the neighborhood of a six-vertex cycle in a P_7-free graph, loosely inspired by [5]. On a high level, we show that there is a branching process with polynomial number of outcomes that in some sense "correctly" completes G to a chordal bipartite graph, proving Theorem 1.3.

To sum up, the proof of Theorem 1.1 consists of three ingredients. First, in Section 3 we show the guesswork that leads to a polynomial number of candidate "almost containers" $(K, \mathcal{D}, L)$ for a minimal separator S in the input graph G. Second, in Section 4 we focus on bipartite graphs and prove Theorem 1.3. Finally, in Section 5, we show how these two tools combine with the dynamic programming framework of [19, 1, 13] to prove Theorem 1.1.

Let us emphasize that even if we aim to solve Feedback Vertex Set in P_7-free graphs of bounded clique number and use our approach to reduce the problem to the bipartite case, the problem we need to solve the latter class is not just Feedback Vertex Set, but a certain extension variant considering CMSO_2 types. Thus even in this case the boundedness of mim-width implied by the work of Lozin and Zamaraev [33] is not sufficient as the known algorithm for bounded-mim-width graphs cannot handle such a problem. Consequently, we need to use Theorem 1.3.

2 Preliminaries

For an integer n, by $[n]$ we denote the set $\{1, 2, \ldots, n\}$.

The set of vertices of graph G is denoted as $V(G)$ and the set of edges as $E(G)$. An edge joining vertices v and u is denoted as uv. By $N(v)$ we denote the set of neighbors of the vertex v (called its *open neighborhood*). By $N[v]$ we denote $N(v) \cup \{v\}$ (called them*closed neighborhood*). Given a set of vertices S, we define its open neighborhood $N(S)$ as $\left(\bigcup_{v \in S} N(v)\right) \setminus S$ and its closed neighborhood $N[S]$ as $\left(\bigcup_{v \in S} N[v]\right)$.

A graph H is an *induced subgraph* of G if it can be obtained by vertex deletions. We do not distinguish between sets of vertices and subgraphs induced by them; in cases when it might cause a confusion we write $G[A]$ for an subgraph induced by $A \subseteq V(G)$. We use $G - A$ as a shorthand for $G[V(G) \setminus A]$.

A *class* of graphs $\mathcal{C}$ is a set of graphs. We say that a graph G is *H-free* if G does not contain any induced subgraph isomorphic to H. A class $\mathcal{C}$ is *H-free* if all graphs $G \in \mathcal{C}$ are H-free. By P_t we denote the path on t vertices. A P_4-free graph is called a *cograph*. By C_t we denote the cycle on t vertices. The *length* of a path or a cycle is the number of edges in it.

Given a graph G and its subset of vertices A, a set of connected components of $G - A$ is denoted as $\mathrm{cc}(G - A)$.

We say that a vertex v is *complete* to a set of vertices B if v is adjacent to all vertices in B. Similarly, a set of vertices A is *complete* to a set of vertices B if every vertex of A is complete to B.

We say that a vertex v is *anticomplete* to a set of vertices B if v has no neighbors in B. Similarly, a set of vertices A is *anticomplete* to a set of vertices B if $N(A) \cap B = \emptyset$.

A *module* M is a non-empty subset of vertices of graph G such that every vertex $v \in V(G) \setminus M$ is adjacent either to all vertices of M or to none of them. A module is called

- *strong* if for any other module M' either $M \subseteq M'$, $M' \subseteq M$ or $M \cap M' = \emptyset$
- *proper* if M is a proper subset of $V(G)$
- *maximal* if it is proper and strong and is not contained in any other proper module

Modules M and M' are called *adjacent* if each vertex of M is complete to M' and $M \cap M' = \emptyset$.

Note that for any graph on at least two vertices the family of maximal modules form a partition of $V(G)$.

Given a graph G and the collection of vertex-disjoint modules M, the *quotient graph G/M* is constructed by replacing each module of M with a single vertex, with the same neighbors as the original vertices of the module.

A set of pairwise non-adjacent vertices in a graph G is called *independent*, while a set of pairwise adjacent vertices is a called *clique*.

A subgraph H of G, which is a connected component in the complement of G, is called an *anti-component*.

2.1 PMCs and chordal completions

A graph G is called *chordal* if all its induced cycles are of length 3. Note that all induced subgraphs of a chordal graph are also chordal.

A graph H is called a *supergraph* of G if $V(H) = V(G)$ and $E(G) \subseteq E(H)$. A graph H is called a *chordal completion* of G if H is a supergraph of G and H is a chordal graph. A chordal completion H of G is called *minimal* if it does not contain any proper subgraph which is also a chordal completion of G. Alternatively, we denote a chordal completion H of G as $G + F$, where F is a set of non-edges in G: then $E(H) = E(G) \cup F$.

A set of vertices $\Omega \subseteq V(G)$ is called a *potential maximal clique* or a *PMC* if there exists a minimal chordal completion H of G in which Ω is a maximal clique.

Let us recall a classic result characterizing potential maximal cliques.

Lemma 2.1 (see [8]). *Given a graph G, a set of vertices Ω is a PMC if and only if the following conditions are fulfilled:*

1. *For each connected component D of $G - \Omega$, the set $N(D)$ is a proper subset of Ω.*
2. *If uv is a non-edge of G, with $u, v \in \Omega$, then there exists a component D of $G - \Omega$ such that $u, v \in N(D)$.*

We say that a set of vertices S is a (u, v)- *separator* if u and v belong to distinct components in $G - S$. A (u, v)-separator S is called *minimal* if no proper subset of S is a (u, v)-separator. A set of vertices S is called a *minimal separator* of graph G if there exists a pair of vertices u and v such that S is a minimal (u, v)-separator.

Given a set of vertices S, we say that a component $D \in \mathrm{cc}(G - S)$ is a *full component* of S if $N(D) = S$. It is easy to prove that S is a minimal separator of G if and only if it has at least two full components. It was proven (see [8]) that if Ω is a PMC in G, then for every component D of $G - \Omega$ the set $N(D)$ is a minimal separator with D as a full component and another full component containing $\Omega \setminus N(D)$.

Bouchitté and Todinca [7, 8] showed a close relation between potential maximal cliques and minimal separators.

Theorem 2.2 (see [7, 8]). *If G is an n-vertex graph with a minimal separators and b potential maximal cliques, then $b \leq n(a^2 + a + 1)$ and $a \leq nb$. Furthermore, given a graph G, one can in time polynomial in the input and output the list of all its minimal separators and potential maximal cliques.*

2.2 Treedepth structures and treewidth

The *rooted forest* $\mathcal{T}$ is a forest in which each component has exactly one distinguished vertex, called a *root*. A path in a rooted forest is called *vertical* if it connects a vertex and any of its ancestors. Given a vertex v, we define its *depth* as the number of vertices on the path connecting v and the root (so the root has depth 1). The *height* of the rooted forest $\mathcal{T}$ is equal to the maximum depth of a vertex of $\mathcal{T}$. By $\mathcal{T}^{\alpha}$ we denote a set of vertices of $\mathcal{T}$ of depth exactly α. We also write $\mathcal{T}^{>\alpha}$ for $\bigcup_{\alpha' > \alpha} \mathcal{T}^{\alpha'}$.

We say that vertices u and v are $\mathcal{T}$*-comparable* if they can be connected via a vertical path. Otherwise, we say that they are $\mathcal{T}$*-incomparable*. An *elimination forest* of graph G is a rooted forest $\mathcal{T}$ such that $V(\mathcal{T}) = V(G)$ and for each edge $uv \in E(G)$ vertices u and v are $\mathcal{T}$-comparable. We define the *treedepth* of graph G as the minimum height of any possible elimination forest of graph G.

Given a graph G and an integer d, a *treedepth-d structure* (the notion first proposed in [13]) is a rooted forest $\mathcal{T}$ of height at most d such that $V(\mathcal{T})$ is a subset of $V(G)$ and $\mathcal{T}$ is an elimination forest of the subgraph of G induced by $V(\mathcal{T})$. A treedepth-d structure $\mathcal{T}$ is a *substructure* of a treedepth-d structure $\mathcal{T}'$ if $\mathcal{T}$ is a subgraph of $\mathcal{T}'$ (as a rooted forest) and every root of $\mathcal{T}$ is also a root of $\mathcal{T}'$. A treedepth-d structure $\mathcal{T}$ is called *maximal* if there is no treedepth-d structure $\mathcal{T}'$ such that $\mathcal{T}$ is a substructure of $\mathcal{T}'$ and $\mathcal{T} \neq \mathcal{T}'$. Equivalently, $\mathcal{T}$ is maximal if it cannot be extended by adding a leaf preserving the bound on the height of $\mathcal{T}$.

To avoid notational clutter, we sometimes treat $\mathcal{T}$ as set of vertices, for example in expressions $|A \cap \mathcal{T}|$.

Given a treedepth-d structure $\mathcal{T}$ of graph G, a chordal completion $G + F$ is $\mathcal{T}$*-aligned* if for any edge $uv \in F$ the following conditions hold:

1. neither u and v is a vertex of depth d in $\mathcal{T}$,
2. if both u and v belong to $V(\mathcal{T})$, then u and v are $\mathcal{T}$-comparable.

The second condition guarantees that $\mathcal{T}$ is a treedepth-d structure in $G + F$. It was proven (see Lemma 2.11 in [13]) that for any treedepth-d structure in graph G, there exists a minimal chordal completion $G + F$ which is $\mathcal{T}$-aligned. We say that a PMC is $\mathcal{T}$*-avoiding* if it is a maximal clique in some minimal chordal completion which is $\mathcal{T}$-aligned and does not contain any vertex of $\mathcal{T}$ of depth d.

Given a treedepth-d structure $\mathcal{T}$, we say that a set S' is a $\mathcal{T}$*-container of defect f* (or a container of defect f if $\mathcal{T}$ is known from the context) for a set S if $S \subseteq S'$ and $|S' \cap V(\mathcal{T})| - |S \cap V(\mathcal{T})| \leq f$. Note we have $S \cap V(\mathcal{T}) \subseteq S' \cap V(\mathcal{T})$, so the defect f describes how many additional vertices of $V(\mathcal{T})$ belong to S'.

Now we can cite the Lemma 2.12 from [13], which shows how to deal with PMCs, which are $\mathcal{T}$-aligned and contain vertices of $\mathcal{T}$ of depth d:

Lemma 2.3 (see Lemma 2.12 in [13]). *For each positive integer d, there is a polynomial-time algorithm which takes in a graph G and returns a collection $L \subseteq 2^{V(G)}$ such that for any maximal treedepth-d structure $\mathcal{T}$ in G, any $\mathcal{T}$-aligned minimal chordal completion $G + F$ of G, and any maximal clique Ω of $G + F$ which contains a depth-d vertex of $\mathcal{T}$, L contains a set $\widetilde{\Omega}$ that is a container for Ω of defect 0, i.e., $\Omega \subseteq \widetilde{\Omega}$ and $\widetilde{\Omega} \cap \mathcal{T} = \Omega \cap \mathcal{T}$.*

In our paper we will also need Lemma 2.13 from [13], which shows how to use the maximality of a treedepth-d structure:

Lemma 2.4 (see Lemma 2.13 in [13]). *Let G be a graph, d be a positive integer, and $\mathcal{T}$ be a maximal treedepth-d structure in G. Then for any $\mathcal{T}$-avoiding potential maximal clique Ω of G, each vertex in $\Omega \setminus \mathcal{T}$ has a neighbor in $\mathcal{T} - \Omega$.*

A *tree decomposition* of a graph G is a pair $\mathcal{T} = (T, \beta)$, where T is a tree and β is a function assigning each node $v \in V(T)$ a subset of vertices $\beta(v) \subseteq V(G)$, called a *bag* of v such that the following conditions are fulfilled:

- for each $u \in V(G)$ a set of nodes of T whose bags contain u induces a connected non-empty subtree of T;
- for each edge $u_1u_2 \in E(G)$ there exists a node $v \in V(T)$ such that $\beta(v)$ contains u_1, u_2.

A *width* of a tree decomposition $\mathcal{T}$ is equal to $\max_{x \in V(T)} |\beta(x)| - 1$. The minimum width over all tree decompositions of a graph G is called the *treewidth* of G and is denoted as $\mathrm{tw}(G)$.

2.3 MWIS

In all problems of interest in this paper, the input graph G comes with vertex weights that are positive rational numbers.

For fixed integer $d \geq 0$ and CMSO_2 formula ϕ with one free vertex set variable, the $(\mathrm{td} \leq d, \phi)$-MWIS problem takes in input a vertex-weighted graph G and asks for a pair (Sol, X) where $X \subseteq \mathsf{Sol} \subseteq V(G)$, $G[\mathsf{Sol}]$ is of treedepth at most d, and $\phi(X)$ is satisfied in $G[\mathsf{Sol}]$, and, subject to the above, the weight of X is maximized (or a negative answer if such a pair does not exist.)

$(\mathrm{td} \leq d, \psi)$-MWIS

Input: A graph G equipped with a weight function $\mathfrak{w} \colon V(G) \to \mathbb{Z}_+$.

Task: Find a pair (Sol, X) such that

- $X \subseteq \mathsf{Sol} \subseteq V(G)$,
- $G[\mathsf{Sol}]$ is of treedepth at most d,
- $G[\mathsf{Sol}] \models \psi(X)$,
- X is of maximum weight subject to the conditions above,

or conclude that no such pair exists.

Similarly as in [13], for a solution (Sol, X) we will be mostly looking at some maximal treedepth-d structure $\mathcal{T}$ that contains Sol.

2.4 Structural lemmas

In this section we present some structural lemmas about P_7-free graphs. Beforehand, we need the following notation.

By $X_1 - X_2 - \cdots - X_k$ we denote a path $v_1 - v_2 - \cdots - v_k$, where v_i is some vertex belonging to X_i for each $i \in [k]$. If X_i is singleton $\{x_i\}$ for some i, we write $-x_i-$ instead $-\{x_i\}-$.

Lemma 2.5. *Let S be a minimal separator in a P_7-free graph G and let A and B be the full components to S. There exists a subset $Z \subseteq A$ such that Z is a cograph and $S \setminus N(Z)$ is complete to B.*

Proof. Let F be a set of vertices of S which are complete to B. Let Z be a minimal connected subgraph of A such that $S \setminus F \subseteq N(Z)$.

Let Q be a maximal induced path in Z and let v be its endpoint. Note that v cannot be a cutvertex of Z – otherwise we could extend our maximal path Q. By the minimality of Z there exists a vertex $w \in S \setminus F$ such that $N(w) \cap Z = \{v\}$. Let B_1 be $N(w) \cap B$ and $B_2 = B \setminus B_1$. As S is the minimal separator and B is a full component to S, the set B_1 is non-empty. Since $w \notin F$, the set B_2 is non-empty. As B is a connected component, then there exist $x_1 \in B_1$, $x_2 \in B_2$, which are adjacent.

Let $P = Q - w - x_1 - x_2$. Note that P is an induced path in G. We assumed that G is P_7-free, thus $|P| < 7$, which implies $|Q| \leq 3$. Thus, Z is a cograph. $\square$

Lemma 2.6 (see Lemma 4.2 in [23]). *Let S be a minimal separator in graph G and let A be a full component to S and $|A| \geq 2$. Let p and q be any two distinct vertices of A belonging to different maximal proper adjacent strong modules of $G[A]$. Then, for each vertex $v \in S$ at least one of the following conditions is fulfilled:*

1. *there exists an induced path $v - A - A - A$*
2. *we have $v \in N(p) \cup N(q)$*
3. *$\overline{G[A]}$ is disconnected and $N(v) \cap A$ consists of some connected components of $\overline{G[A]}$*

Corollary 2.7 (compare Lemma 8 in [38]). *Let S be a minimal separator in graph G and let A and B be full components to S. There exists a set $X \subseteq A$ ($X \subseteq B$) of size at most $\omega(G)$ such that for each vertex v belonging to $S \setminus N(X)$ there exists an induced path $v - A - A - A$ ($v - B - B - B$).*

Proof. If $|A| < 2$, then we simply take $X = A$. Suppose then that A consists of least two vertices. Let Q be a set of such vertices $v \in S$ such that there exists an induced path $v - A - A - A$. If the complement of $G[A]$ is disconnected, then we choose one vertex from each connected component of $\overline{G[A]}$ and create a set X from the chosen vertices. If $\overline{G[A]}$ is connected, let us pick any two vertices belonging to different maximal proper adjacent modules of $G[A]$ and construct X this way. In both cases we have $|X| \leq \omega(G)$. By Lemma 2.6 for any vertex $v \in S \setminus N(X)$ we have $v \in Q$. We prove the statement for B in symmetrical way. $\square$

We will also need the following lemma, which was proved in [23].

Lemma 2.8 (see Lemma 4.1 in [23]). *Let $(X, \leq_1)$ and $(X, \leq_2)$ be two partial orders on the set X such that any pair of elements of X is comparable in $\leq_1$ or in $\leq_2$. Then, there exists an element v such that for any element $x \in X$ we have $v \leq_1 x$ or $v \leq_2 x$.*

We say that sets A and B are *comparable by inclusion* if $A \subseteq B$ or $B \subseteq A$.

Lemma 2.9. *Let G be a P_7-free graph and let Z be an induced subgraph of G, which is a connected cograph. Let $I \subseteq V(G) \setminus N[Z]$ be an independent set and let $J \subseteq N(Z) \cap N(I)$ an independent set. Then there exist sets of vertices $Q_Z \subseteq Z$ and $Q_I \subseteq I$ such that $J \subseteq N(Q_Z) \cup N(Q_I)$ and $|Q_Z|, |Q_I| \leq (\omega(Z) + 1)!$.*

Proof. Let $f\colon \mathbb{N} \to \mathbb{N}$ be defined recursively as $f(1) = 1$ and $f(k) \leq k(1 + f(k-1))$ for $k \geq 2$. A straightforward induction shows that for every $k \in \mathbb{N}$ we have $f(k) \leq (k+1)!$.

We will prove the lemma by the induction on $\omega(Z)$ and with the bound $|Q_I|, |Q_Z| \leq f(\omega(Z))$. For the base case, if $\omega(Z) = 1$, then Z contains only one vertex v and $J \subseteq N(v)$. Thus, the induction holds for $\omega(Z) = 1$.

Let us assume now that $\omega(Z) > 1$. Let $Z_1, \ldots, Z_l$ be anticomponents of Z. Note that $l \leq \omega(Z)$. Since Z is a connected cograph with $\omega(Z) > 1$, we know that $l > 1$. For each $i \in [l]$ we choose an arbitrary vertex $z_i \in Z_i$. Let $\{Z_i^1, Z_i^2, \ldots, Z_i^{m_i}\}$ be a set of connected components of $Z_i - z_i$.

Let $J' = J \setminus N(\{z_1, \ldots, z_l\})$. We split now J' depending on the neighborhood in Z. Let $J'_i = (J' \cap N(Z_i)) \setminus N(Z_1 \cup \cdots \cup Z_{i-1})$.

Suppose that $m_i \geq 2$. Let us consider two partial orders on J'_i:

- $\leq_1$: $u \leq_1 v \Leftrightarrow N(u) \cap I \subseteq N(v) \cap I$
- $\leq_2$: $u \leq_2 v \Leftrightarrow \{j \in [m_i] \mid u \in N(Z_i^j)\} \subseteq \{j \in [m_i] \mid v \in N(Z_i^j)\}$

Suppose that there exist vertices $x, y \in J_i'$ which are incomparable in both $\leq_1$ and $\leq_2$. Then there exists vertices $x_i, y_i \in I$ such that $x_i x, y_i y \in E(G)$, but $x y_i, x_i y \notin E(G)$. Similarly, there exist connected components $Z_i^{j_x}$ and $Z_i^{j_y}$ such that $x \in N(Z_i^{j_x})$ and $y \in N(Z_i^{j_y})$, but $y \notin N(Z_i^{j_x})$ and $x \notin N(Z_i^{j_y})$. So there exist induced paths P_x and P_y connecting x and respectively y with $z_{i'}$, where $i' \in [l] \setminus \{i\}$ such that $P_x \subseteq V(Z_i^{j_x})$ and $P_y \subseteq V(Z_i^{j_y})$. Since $x, y \notin N(z_{i'})$, each of P_x and P_y must contain at least one vertex. Then a path $x_i - x - P_x - z_{i'} - P_y - y - y_i$ is an induced path of at least 7 vertices, a contradiction.

Thus each pair of elements of J_i' is comparable in $\leq_1$ or $\leq_2$. Then by Lemma 2.8 there exists a vertex $w_i \in J_i'$ such that any other vertex $v \in J_i'$ we have $w_i \leq_1 v$ or $w_i \leq_2 v$. Let $u_{w_i}^i$ be any vertex in $N(w_i) \cap I$ – such vertex exists because $J \subseteq N(I)$. Let j_{w_i} be any index such that $N(w_i) \cap Z_i^{j_{w_i}} \neq \emptyset$ (such component exists, because of the definition of J_i and the assumption $m_i \geq 2$). Then $J_i' \subseteq N(u_{w_i}^i) \cup N(Z_i^{j_{w_i}})$. Since Z_i is the anticomponent, $\omega(Z_i^{j_{w_i}}) \leq \omega(Z_i) \leq \omega(Z) - 1$. By the inductive assumption, applied to a connected cograph $Z_i^{j_{w_i}}$, $J_i' \setminus N(u_{w_i}^i)$ and I, there exist sets Q_Z^i and Q_I^i of size at most $f(\omega(Z) - 1)$ such that $J_i' \setminus N(u_{w_i}^i) \subseteq N(Q_Z^i) \cup N(Q_I^i)$.

We are left with cases $m_i = 0$ and $m_i = 1$. In the former, $Z_i = \{z_i\}$ and $J_i' = \emptyset$. We take then $Q_Z^i = Q_I^i = \emptyset$. If $m_i = 1$, then $J_i' \subseteq N(Z_i^1)$. We can use directly inductive assumption to Z_i^1, J_i' and I, getting sets Q_Z^i and Q_I^i such that $J_i \subseteq N(Q_Z^i) \cup N(Q_I^i)$.

Let $Q_Z = \{z_1, \ldots, z_l\} \cup \bigcup_{i \in [l]} Q_Z^i$ and $Q_I = \{u_{w_1}^1, \ldots, u_{w_l}^l\} \cup \bigcup_{i \in [l]} Q_I^i$. Then

$$|Q_Z|, |Q_I| \leq \omega(Z) + \omega(Z) \cdot f(\omega(Z) - 1) = f(\omega(Z)).$$

We also have

$$J \subseteq N(\{z_1, \ldots, z_l\}) \cup J' = N(\{z_1, \ldots, z_l\}) \cup \bigcup_{i \in [l]} J_i' \subseteq N(Q_Z) \cup N(Q_I).$$

□

2.5 χ-boundness

By $\omega(G)$ we denote the *clique number* of G, i.e. a size of the maximal clique being a subgraph of G. By $\chi(G)$ we denote the *chromatic number* of graph G, i.e. the minimal number of colors required to properly color vertices of graph G (so no two adjacent vertices get the same color assigned). For any graph G we have $\omega(G) \leq \chi(G)$. There exist graphs with large girth and large chromatic number, so we cannot hope for bound $\chi(G) \leq f(\omega(G))$ for any graph G. This leads us to the notion of *χ-boundness*, which was discussed by Gyárfás in [25]. We say that a class $\mathcal{C}$ is *χ-bounded* if there exists a function $f \colon \mathbb{N} \to \mathbb{N}$ such that for any graph $G \in \mathcal{C}$ we have $\chi(G) \leq f(\omega(G))$.

Gyárfás proved in [25] the following theorem:

Theorem 2.10 (Theorem 2.4 in [25])**.** *Class of graphs excluding P_t is χ-bounded and a function $f_n(x) = (t-1)^{x-1}$ is a suitable bounding function.*

Proof of this theorem, presented in [25], is constructive and it shows that a P_t-free graph G can be colored with $(t-1)^{\omega(G)-1}$ colors (i.e. divided into color classes) in polynomial time.

2.6 2VC-dimension

The notion of Vapnik-Chervonenkis dimension was introduced by Vapnik and Chervonenkis in [39]. In graphs it is usually applied with the set of vertices as a universe and the family of closed neighborhoods as family of sets. Here, we will use VC-dimension in a slightly different setting.

Let $(U, \mathcal{F})$ be a *set system*: let U be a set and $\mathcal{F}$ be a family of subsets of U. The *2VC-dimension* of a set system $(U, \mathcal{F})$, denoted as $\mathrm{2VCdim}(U, \mathcal{F})$ is a maximal size of a set $X \subseteq U$ such that for every subset $Y \subseteq X$ of size 2 there exists $A \in \mathcal{F}$ such that $A \cap X = Y$.

For a set system $(U, \mathcal{F})$, we define a *dual set system* as $(\mathcal{F}, \widehat{U})$, where $\widehat{U} = \{x^* \mid x \in U\}$ and $x^* = \{F \in \mathcal{F} \mid x \in F\}$. Then a *dual 2VC-dimension* of set system $(U, \mathcal{F})$, denoted as $\mathrm{2VCdim}^*(U, \mathcal{F})$, is a 2VC-dimension of its dual system, i.e. the size of a maximal $\mathcal{F}' \subseteq \mathcal{F}$ such that for every distinct $A, B \in \mathcal{F}'$ there exists $u \in U$ such that $\{C \in \mathcal{F}' \mid u \in C\} = \{A, B\}$.

Given a set system $(U, \mathcal{F})$, let $\nu(U, \mathcal{F})$ be a maximal size of a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ of pairwise disjoint sets and let $\tau(U, \mathcal{F})$ be a minimal size of a set $A \subseteq U$ such that for every $F \in \mathcal{F}$ we have $F \cap A \neq \emptyset$.

Ding, Seymour and Winkler proved the lemma in [17], bounding τ as a function of λ and ν. They used the language of hypergraphs, which we translated here to set systems.

Lemma 2.11 (compare (1.1) in [17]). *For any set system $\mathcal{S} = (U, \mathcal{F})$ we have*

$$\tau(\mathcal{S}) \leq 11 \cdot \mathrm{2VCdim}^*(\mathcal{S})^2 \left(\mathrm{2VCdim}^*(\mathcal{S}) + \nu(\mathcal{S}) + 3\right) \binom{\mathrm{2VCdim}^*(\mathcal{S}) + \nu(\mathcal{S})}{\nu(\mathcal{S})}^2 .$$

Corollary 2.12. *For every t there exists c such that for every P_t-free graph G, a PMC Ω in G and an independent set $I \subseteq \Omega$ such that for every vertex $v \in I$ there exists a component $D \in \mathrm{cc}(G - \Omega)$ such that $v \in N(D)$, there exists a family $\mathcal{D} \subseteq \mathrm{cc}(G - \Omega)$ of size at most c such that $I \subseteq \bigcup_{D \in \mathcal{D}} N(D)$.*

Proof. Let us consider a set system $\mathcal{S} = (\mathrm{cc}(G - \Omega), \{\mathcal{D}_v \mid v \in I\})$, where $\mathcal{D}_v = \{D \in \mathrm{cc}(G - \Omega) \mid v \in N(D)\}$. By Lemma 2.1 we know that for every two non-adjacent vertices $u, v \in \Omega$ there exists a component $D \in \mathrm{cc}(G - \Omega)$ such that $u, v \in N(D)$. Hence, $\nu(\mathcal{S}) = 1$, as I is an independent set. Let us denote $\left\lceil \frac{t}{2} \right\rceil + 1$ as p. We will show now that $\mathrm{2VCdim}^*(\mathcal{S}) < p$. Suppose otherwise – then there exists a subfamily $\mathcal{F}' = \{\mathcal{D}_{v_1}, \mathcal{D}_{v_2}, \dots, \mathcal{D}_{v_p}\}$ such that for every $j, k \in \left[\frac{t}{2}\right]$ there exists a component $D \in \mathrm{cc}(G - \Omega)$ such that $\{\mathcal{D}_i \in \mathcal{F}' \mid D \in \mathcal{D}_i\} = \{\mathcal{D}_{v_i}, \mathcal{D}_{v_j}\}$. In other words, for every two distinct vertices $u, w \in \{v_1, \dots, v_p\} \subseteq I$ there exists a component $D \in \mathrm{cc}(G - \Omega)$ such that $u, w \in N(D)$, but no other $v_i \in N(D)$ and therefore a minimal path $P_{uw} \subseteq D$ connecting u and w (note that P_{uw} may contain just one vertex). Let us consider a path $v_1 - P_{v_1 v_2} - v_2 - P_{v_2 v_3} - v_3 - \cdots - P_{v_{p-1} v_p} - v_p$. Note that it is an induced path in G of length $p + p - 1 \geq t$, which is a contradiction. Thus, $\mathrm{2VCdim}^*(\mathcal{S}) < p$.

Note that $\tau(\mathcal{S})$ is a minimal size of a set $\mathcal{D} \subseteq \mathrm{cc}(G - \Omega)$ such that for every $v \in I$ there exists $D \in \mathcal{D}$ such that $v \in N(D)$. By Lemma 2.11 we get that $\tau(S)$ is bounded by a function of p, so a function of t. ☐

We immediately get the following corollary.

Corollary 2.13. *For every t and k there exists c such that for every P_t-free graph G with $\omega(G) \leq k$ and a PMC Ω one of the two following conditions hold:*

1. *there exists $v \in \Omega$ such that for every component $D \in \mathrm{cc}(G - \Omega)$ we have $v \notin N(D)$*
2. *there exists a family $\mathcal{D} \subseteq \mathrm{cc}(G - \Omega)$ of size at most c such that $\Omega = \bigcup_{D \in \mathcal{D}} N(D)$*

3 Approximating minimal separators

This section is devoted to the proof of the following theorem.

Theorem 3.1. *Let $d, k \geq 1$ be fixed integers. Given a P_7-free graph G with $\omega(G) \leq k$, one can in polynomial time compute a family $\mathcal{F}$ with the following guarantee: for every maximal treedepth-d structure $\mathcal{T}$ and every $\mathcal{T}$-avoiding minimal separator S in G with full components A and B, there exists $(K, \mathcal{D}, L) \in \mathcal{F}$ satisfying the following:*

- *$K \subseteq V(G)$, $S \cap \mathcal{T} \subseteq K$, and $|K \cap \mathcal{T}| = \mathcal{O}_{k,d}(1)$;*
- *$\mathcal{D}$ is a subset of the family of connected components of $G - K$;*
- *L is a function that assigns to every $D \in \mathcal{D}$ a subset $\emptyset \neq L(D) \subsetneq D$ such that every connected component of either $G[L(D)]$ or of $G[D \setminus L(D)]$ is a module of $G[D]$;*

- *for every connected component D of $G - K$, one of the following holds:*
  - *D is a subset of a single connected component of $G - S$; or*
  - *$D \in \mathcal{D}$.*

Proof. Let G be as in the theorem statement. Using Theorem 2.10, we compute a coloring of G into $c \leq 6^{k-1}$ color classes and denote them by $V^1, \ldots, V^c$. In what follows, we will frequently divide various subsets of $V(G)$ into color classes; for a set X, we usually denote $X \cap V^i$ by X^i for brevity and call the sets $X^1, \ldots, X^c$ the color classes of X.

Let $\mathcal{T}$ be a maximal treedepth-d structure in G. Let S be a $\mathcal{T}$-avoiding minimal separator in graph G and let A and B be the full components to S. We follow here a way from presentation from [1, 13]: we want to guess the tuple $(K, \mathcal{D}, L)$ for (S, A, B) by making a guesswork that ends up in a polynomial number of options. That is, we describe a nondeterministic polynomial-time algorithm that

1. has a polynomial number of possible runs,
2. in every run, the algorithm either outputs a tuple $(K, \mathcal{D}, L)$ as in the lemma statement or terminates without outputting any tuple, and
3. for every $\mathcal{T}$, S, A, and B as above, there exists a run of the algorithm that produces a tuple $(K, \mathcal{D}, L)$ fitting $\mathcal{T}$, S, A, and B as in the lemma statement.

To compute the desired family $\mathcal{F}$, we iterate over all possible runs of the algorithm and collect all output tuples $(K, \mathcal{D}, L)$. The nondeterministic algorithm does not know $\mathcal{T}$, S, A, nor B, but is able to nondeterministically guess some properties of them; however, in such guesses we need to control the number of possible runs (branches) so that their total number is polynomial. For example, while we cannot guess the entire set S, we can for example guess one leaf of $\mathcal{T}$ that is contained in A (there are n options) or a specific vertex of S.

Let X_A be a subset of A of size at most k such that for each vertex $v \in S \setminus N(X_A)$ there exists an induced path $v - A - A - A$. Analogously, we define X_B as a subset of B of size at most k such that for every vertex $v \in S \setminus N(X_B)$ there exists an induced path $v - B - B - B$. Such subsets exists by Corollary 2.7.

Suppose that there exists $v \in S \setminus (N(X_A) \cup N(X_B))$. Then v belongs to $S \setminus N(X_B)$ and $S \setminus N(X_A)$, so there exists induced paths $v - B - B - B$ and $v - A - A - A$. By joining them, we create a P_7 which is a contradiction. Thus, $S \subseteq N(X_A) \cup N(X_B)$. Moreover, note that $N(X_A) \cap N(X_B) \subseteq S$.

We will maintain a set $\widetilde{S}$ which will serve as our current candidate for K. We initiate $\widetilde{S} = \emptyset$. We will refer to $\widetilde{S}$ as "our container" for the set S.

We (nondeterministically) guess sets X_A, X_B and leafs p_A and p_B of $\mathcal{T}$, belonging to respectively A and B. By Lemma 2.4 such vertices exists, as we are guessing vertices of maximal depth in $\mathcal{T} \cap A$ and $\mathcal{T} \cap B$. We also know that there exists suitable guesses for X_A and X_B, fulfilling the above definitions. Since $|X_A|, |X_B| \leq k$, there is a polynomial number of options for this guess.

Then we add to $\widetilde{S}$ vertices of $N(X_A \cup \{p_A\}) \cap N(X_B \cup \{p_B\})$ and $N(p_A) \cup N(p_B)$. We also guess $\mathcal{T} \cap S$ and add them to $\widetilde{S}$. Since S is $\mathcal{T}$-avoiding, $|\mathcal{T} \cap S| \leq d - 1$.

Note that we added to $\widetilde{S}$ at most $d - 1$ ancestors of p_A among vertices in $N(p_A)$ and at most $d - 1$ ancestors of p_B among vertices in $N(p_B)$ in the treedepth-d structure.

Let $S_A = S \setminus N(X_A)$ and $S_B = S \setminus N(X_B)$. Note that S_A and S_B are subsets of respectively $N(X_B)$ and $N(X_A)$.

We now focus on vertices of S, which have neighbors in connected components others than A and B. Let $\widehat{S} = N(V(G) \setminus (A \cup B \cup S))$, note that $\widehat{S} \subseteq S$, and let $\widehat{S_A} = \widehat{S} \cap S_A$ and $\widehat{S_B} = \widehat{S} \cap S_B$. We divide $\widehat{S_A}$ and $\widehat{S_B}$ into color classes: $\widehat{S^1_A}, \widehat{S^2_A}, \ldots, \widehat{S^c_A}$ and $\widehat{S^1_B}, \widehat{S^2_B}, \ldots, \widehat{S^c_B}$. We need the following lemma in order to put these vertices into our container.

Claim 3.2. *For any* $i \in [c]$, *if* $\widehat{S^i_A} \neq \emptyset$, *then there exists a component* $D^i_A \in \mathrm{cc}(G - (A \cup S \cup B))$ *and vertices* $a^i_A, b^i_A, c^i_A \in A$ *such that a set of vertices* $\widehat{S^i_A}$ *is a subset of* $N(D^i_A) \cup (N(\{a^i_A, b^i_A, c^i_A\}) \cap N(X_B))$. *A symmetrical statement holds for* B.

Proof. Let v be a vertex in $\widehat{S^i_A}$ which has the neighbors in the least number of connected components of $G - (A \cup S \cup B)$. Then there exists an induced path $v - A - A - A$ – let a^i_A, b^i_A, c^i_A be vertices of A such that $v - a^i_A - b^i_A - c^i_A$ is an induced path. Let D^i_A be any connected component of $G - (A \cup S \cup B)$ such that $v \in N(D^i_A)$.

Suppose that there exists a vertex $u \in \widehat{S^i_A}$ which does not belong to a set $N(D^i_A) \cup (N(\{a^i_A, b^i_A, c^i_A\}) \cap N(X_B))$. As $\widehat{S^i_A}$ is a color class, u and v are non-adjacent. However, we can connect them via B, as each vertex in S_A has a neighbor in X_B and B is a connected component. Let $Q \subseteq B$ be a path (maybe on just one vertex) such that $u - Q - v$ is an induced path in G. By the minimality of v, we know that there exists a connected component $D' \in \mathrm{cc}(G - (A \cup S \cup B))$ and a vertex $w \in D'$ such that $v \notin N(D')$ and u and w are adjacent. Then $w - u - Q - v - a^i_A - b^i_A - c^i_A$ is an induced path on at least seven vertices; a contradiction. Thus, $\widehat{S^i_A} \subseteq N(D^i_A) \cup (N(\{a^i_A, b^i_A, c^i_A\}) \cap N(X_B))$.

The proof for B is symmetrical. ☐

Note that each component $D \in \mathrm{cc}(G - (A \cup S \cup B))$ is a connected component of $G - S$, different from A and B. Since $S \subseteq N(X_A) \cup N(X_B) \subseteq A \cup S \cup B$, then $D \in \mathrm{cc}(G - (N(X_A) \cup N(X_B)))$. Therefore each connected component of $G - (A \cup S \cup B)$ can be guessed.

Now we use Claim 3.2 to expand $\widetilde{S}$ and therefore to cover $\widehat{S}$. For each $\widehat{S^i_A}$ such that $i \in [c]$ we guess if $\widehat{S^i_A} \neq \emptyset$ and, if this is the case, we guess a connected component D_i in $G - (N(X_A) \cup N(X_B))$ and vertices a^i, b^i, c^i and we add $N(D_i) \cup \big(N(\{a^i, b^i, c^i\}) \cap N(X_B)\big)$ to $\widetilde{S}$. We do symmetric guesses for each $\widehat{S^i_B}$ for each $i \in [c]$.

At this point, all vertices of $S \setminus \widetilde{S}$ have neighbors only in $A \cup S \cup B$. We will use the following two claims to cover some vertices which have two incomparable neighbors in $\mathcal{T}$.

Fix $\alpha \in [d]$. Let S^α_A be a subset of $S_A \setminus \widetilde{S}$, consisting of these vertices of S_A which have at least two neighbors in $\mathcal{T}^\alpha \cap B$. Let us divide S^α_A into color classes $S^{\alpha,1}_A, S^{\alpha,2}_A, \ldots, S^{\alpha,c}_A$. Similarly, let S^α_B be a subset of $S_B \setminus \widetilde{S}$, consisting of these vertices of S_B which have two neighbors in $\mathcal{T}^\alpha \cap A$. Again, we divide S^α_B into color classes $S^{\alpha,1}_B, S^{\alpha,2}_B, \ldots, S^{\alpha,c}_B$.

Claim 3.3. *For each* $\alpha \in [d]$ *and* $i \in [c]$ *there exist* $Q_Z \subseteq A$ *of size at most* $2(k+1)!$, *and disjoint sets of vertices* $Q^1_I, Q^2_I \subseteq \mathcal{T}^\alpha \cap B$ *of size at most* $(k+1)!$ *each, such that*

$$S^{\alpha,i}_A \subseteq (N(Q_Z) \cap N(X_B)) \cup \left(\bigcup_{x \in Q^1_I} \; \bigcup_{y \in (Q^1_I \cup Q^2_I) \setminus \{x\}} N(x) \cap N(y) \right).$$

A symmetrical statement holds for $S^{\alpha,i}_B$ *for each* $\alpha \in [d]$ *and* $i \in [c]$.

Proof. Let us fix $\alpha \in [d]$ and $i \in [c]$. We consider $S^{\alpha,i}_A$, for shorthand denoted here as J. Since J is a color class, it is an independent set. By Lemma 2.5 there exists a connected cograph Z in A such that $S \setminus N(Z)$ is complete to B. Therefore $J \subseteq N(Z)$ (vertices belonging to S which are complete to B belong to $N(p_B)$, so they belong to $\widetilde{S}$). Let I^1 be set of vertices of $\mathcal{T}^\alpha$ belonging to B. Note that I^1 is an independent set. By Lemma 2.9 there exist sets $Q^1_I \subseteq I^1$ and $Q^1_Z \subseteq Z$ of size at most $(k+1)!$ each such that $J \subseteq N(Q^1_I) \cup N(Q^1_Z)$. Note that $N(Q^1_Z) \cap S^{\alpha,i}_A \subseteq N(Q^1_Z) \cap N(X_B)$.

Let us denote $N(Q^1_Z) \cap N(X_B) \cup \bigcup_{x,y \in Q^1_I,\, x \neq y} N(x) \cap N(y)$ as X. Let $J^2 = J \setminus X$ and $I^2 = I^1 \setminus Q^1_I$. Note that if $v \in J^2$, then v has exactly one neighbor in Q^1_I, so it has a neighbor in I^2. We again use Lemma 2.9, there exists sets $Q^2_I \subseteq I^2$ and $Q^2_Z \subseteq Z$ of size at most $(k+1)!$ such that $J^2 \subseteq N(Q^2_I) \cup N(Q^2_Z)$.

Then we have $J^2 \subseteq (N(Q_Z^2) \cap N(X_B)) \cup (\bigcup_{x \in Q_I^1} \bigcup_{y \in Q_I^2} N(x) \cap N(y))$. Thus J is indeed a subset of

$$S_A^{\alpha,i} \subseteq \left(\left(N(Q_Z^1 \cup Q_Z^2) \cap N(X_B)\right) \cup \left(\bigcup_{x \in Q_I^1} \bigcup_{y \in (Q_I^1 \cup Q_I^2) \setminus \{x\}} N(x) \cap N(y) \right) \right).$$

Setting $Q_Z = Q_Z^1 \cup Q_Z^2$ concludes the proof; the proof for B is symmetric. □

For every $\alpha \in [d]$ and $i \in [c]$ we guess if $S_A^{\alpha,i}$ of S_A^{α} is nonempty, and if it is the case, then we guess sets Q_Z^i, $Q_I^{1,i}$, $Q_I^{2,i}$. By Claim 3.3 we know that there exist suitable guesses such that $S_A^{\alpha,i}$ is contained in

$$\left(\left(N(Q_Z^i) \cap N(X_B)\right) \cup \left(\bigcup_{x \in Q_I^{1,i}} \bigcup_{y \in (Q_I^{1,i} \cup Q_I^{2,i}) \setminus \{x\}} N(x) \cap N(y) \right) \right),$$

which we denote as $X^{\alpha,i}$ (note that X_B is already guessed). Then for each $\alpha \in [d]$ and $i \in [c]$ we add $X^{\alpha,i}$ to $\widetilde{S}$.

We need to bound how many vertices of $\mathcal{T}$ are added to $\widetilde{S}$. Per each $X^{\alpha,i}$ we may have added to $\widetilde{S}$ vertices from $\mathcal{T}$, which are ancestors of two vertices belonging to $Q_I^{1,i}$ and $Q_I^{2,i}$. So we added at most $d \cdot \left| Q_I^{1,i} \cup Q_I^{2,i} \right|^2$. So the number of vertices of $\mathcal{T}$ in $\widetilde{S}$ increased by at most $d \cdot c \cdot d \cdot 4((k+1)!)^2$.

We follow the same steps for each color class $S_B^{\alpha,i}$ of S_B^{α}, guessing the sets and putting vertices into $\widetilde{S}$. Symmetrically, this step increases the number of vertices of $\mathcal{T}$ in $\widetilde{S}$ by at most $4cd^2((k+1)!)^2$.

We put into $\widetilde{S}$ vertices of S_A (resp. S_B), which have at least two neighbors on the same level in $\mathcal{T} \cap B$ (resp. $\mathcal{T} \cap A$). We will use now a similar idea to put into $\widetilde{S}$ some vertices of S which have two incomparable neighbors in $\mathcal{T}$ of two different levels.

Fix distinct $\alpha, \beta \in [d]$. Let $S_A^{\alpha,\beta}$ be a subset of $S_A \setminus \widetilde{S}$ consisting of these vertices of S_A which have two incomparable neighbors in $\mathcal{T} \cap B$ – one of depth α in $\mathcal{T}$ and another of depth β in $\mathcal{T}$. Note that each vertex $v \in S_A^{\alpha,\beta}$ has exactly one vertex in $\mathcal{T}^{\alpha} \cap B$ and exactly one in $\mathcal{T}^{\beta} \cap B$, as vertices having at least two neighbors in either of these two sets already belong to $\widetilde{S}$. We divide each $S_A^{\alpha,\beta}$ into color classes $S_A^{\alpha,\beta,1}, S_A^{\alpha,\beta,2}, \ldots, S_A^{\alpha,\beta,c}$. Let us define $S_B^{\alpha,\beta,i}$ in a symmetrical way.

Claim 3.4. *For each distinct $\alpha, \beta \in [d]$ and $i \in [c]$ there exist a set $Q_Z \subseteq A$ of size at most $2(k+1)!$, and sets Q_I^1 and Q_I^2, which are subsets of respectively $\mathcal{T}^{\alpha} \cap B$ and $\mathcal{T}^{\beta} \cap B$ of size at most $(k+1)!$ such that*

$$S_A^{\alpha,\beta,i} \subseteq (N(Q_Z) \cap N(X_B)) \cup \bigcup_{\substack{x \in Q_I^1 \\ x,y \text{ incomparable in } \mathcal{T}}} \bigcup_{y \in Q_I^2} N(x) \cap N(y)$$

A symmetric statement holds for $S_B^{\alpha,\beta,i}$ for each distinct $\alpha, \beta \in [d]$ and $i \in [c]$.

Proof. Let us fix distinct $\alpha, \beta \in [d]$ and $c \in [c]$. We consider $S_A^{\alpha,\beta,i}$, for short denoted here as J. By Lemma 2.5, there exists a connected cograph Z in A such that $S \setminus N(Z)$ is complete to B. Therefore $J \subseteq N(Z)$. We apply Lemma 2.9 to Z, J and $\mathcal{T}^{\alpha} \cap B$, getting sets Q_Z^1 and Q_I^1 such that $S_A^{\alpha,\beta,i} \subseteq N(Q_Z^1) \cup N(Q_I^1)$. We also apply Lemma 2.9 to Z, J and $\mathcal{T}^{\beta} \cap B$, getting sets Q_Z^2 and Q_I^2 such that $S_A^{\alpha,\beta,i} \subseteq N(Q_Z^2) \cup N(Q_I^2)$. Let $Q_Z = Q_Z^1 \cup Q_Z^2$.

Note that

$$S_A^{\alpha,\beta,i} \setminus N(Q_Z) \subseteq \{N(x) \cap N(y) \mid x \in Q_I^1,\ y \in Q_I^2,\ x \text{ and } y \text{ are incomparable in } \mathcal{T}\},$$

which holds by the definition of $S_A^{\alpha,\beta,i}$. As $N(Q_Z^1 \cup Q_Z^2) \cap S_A^{\alpha,\beta,i} \subseteq N(Q_Z) \cap N(X_B)$, we get

$$S_A^{\alpha,\beta,i} \subseteq (N(Q_Z) \cap N(X_B)) \cup \bigcup_{\substack{x \in Q_I^1 \\ x,y \text{ incomparable}}} \bigcup_{y \in Q_I^2} N(x) \cap N(y)$$

The symmetrical proof holds for B. $\square$

For each $\alpha, \beta \in [d]$ and $i \in [c]$ we guess id $S_A^{\alpha,\beta,i}$ is nonempty, and if this is the case, we guess sets Q_Z, Q_I^1 and Q_I^2. By Claim 3.4 we know that there exists suitable guesses such that $S_A^{\alpha,\beta,i}$ is contained in

$$(N(Q_Z) \cap N(X_B)) \cup \bigcup_{\substack{x \in Q_I^1 \\ x,y \text{ incomparable}}} \bigcup_{y \in Q_I^2} N(x) \cap N(y)$$

which we denote as $X^{\alpha,\beta,i}$. For each distinct $\alpha, \beta \in [d]$ and $i \in [c]$ we add $X^{\alpha,\beta,i}$ to $\widetilde{S}$.

With each $X^{\alpha,\beta,i}$ we may have added at most $d \cdot |Q_I^1 \cup Q_I^2|^2$ vertices from $\mathcal{T}$ to $\widetilde{S}$, so the size of $\mathcal{T} \cap \widetilde{S}$ increases by at most $d \cdot d \cdot c \cdot d(2(k+1)!)^2$ after adding all $X^{\alpha,\beta,i}$'s.

We follow the same steps for $S_B^{\alpha,\beta,i}$, guessing the sets and adding vertices to $\widetilde{S}$. The increase of $|\mathcal{T} \cap \widetilde{S}|$ is bounded similarly.

At this moment, for every vertex $v \in S_A \setminus \widetilde{S}$ there are no incomparable vertices in $N(v) \cap \mathcal{T} \cap B$. A symmetrical statement holds for vertices belonging to $S_B \setminus \widetilde{S}$.

Fix $\alpha \in [d]$. Let us denote set $\{v \in S_A \setminus \widetilde{S} \mid N(v) \cap A \cap \mathcal{T}^\alpha \neq \emptyset\}$ as $\overline{S_A^\alpha}$. Let us divide $\overline{S_A^\alpha}$ into color classes: $\overline{S_A^{\alpha,1}}, \overline{S_A^{\alpha,2}}, \ldots, \overline{S_A^{\alpha,c}}$. Similarly, let $\overline{S_B^\alpha} = \{v \in S_B \setminus \widetilde{S} \mid N(v) \cap B \cap T^\alpha \neq \emptyset\}$ and let $\overline{S_B^{\alpha,1}}, \overline{S_B^{\alpha,2}}, \ldots, \overline{S_B^{\alpha,c}}$ be color classes of $\overline{S_B^\alpha}$.

Claim 3.5. *For each $i \in [c]$ and $\alpha \in [d]$, if $\overline{S_A^{\alpha,i}} \neq \emptyset$, there exists vertices $p^{\alpha,i}, p_1^{\alpha,i}, p_2^{\alpha,i} \in \mathcal{T}^\alpha \cap A$ and vertices $a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i} \in A$ inducing a P_3 such that every $v \in \overline{S_A^{\alpha,i}}$ is either adjacent to at least one of the vertices $p^{\alpha,i}, p_1^{\alpha,i}, p_2^{\alpha,i}, a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}$ or satisfies the following: every $u \in N(v) \cap A \cap \mathcal{T}^\alpha$ has a neighbor among vertices $a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}$.*

A symmetrical statement holds for $\overline{S_B^{\alpha,i}}$ for each $\alpha \in [d]$ and $i \in [c]$.

Proof. Let us fix $i \in [c]$ and $\alpha \in [d]$. We consider $\overline{S_A^{\alpha,i}}$, for short denoted here as J. Let $v^{\alpha,i} \in J$ be a vertex, minimizing $|N(v^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha|$. By the definition of S_A, there exists an induced path $v^{\alpha,i} - a^{\alpha,i} - b^{\alpha,i} - c^{\alpha,i}$, where $a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i} \in A$. Let $p^{\alpha,i}$ be any vertex belonging to $N(v^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha$. Let

$$J_1 = \{v \in J \setminus (N(a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}, p^{\alpha,i}) \cap N(X_B)) \mid N(v) \cap N(v^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha \neq \emptyset\}.$$

Let us consider two partial orders on vertices of J_1:

- $\leq_1$: $u \leq_1 w \Leftrightarrow \big((N(u) \cap A \cap \mathcal{T}^\alpha) \cap N(v^{\alpha,i})\big) \subseteq \big((N(w) \cap A \cap \mathcal{T}^\alpha) \cap N(v^{\alpha,i})\big)$
- $\leq_2$: $u \leq_2 w \Leftrightarrow \big((N(u) \cap A \cap \mathcal{T}^\alpha) \setminus N(v^{\alpha,i})\big) \subseteq \big((N(w) \cap A \cap \mathcal{T}^\alpha) \setminus N(v^{\alpha,i})\big)$

Note that for every $w \in J_1$ the set $\big((N(w) \cap A \cap \mathcal{T}^\alpha) \cap N(v^{\alpha,i})\big)$ is non-empty by the definition of J_1 and the set $\big((N(w) \cap A \cap \mathcal{T}^\alpha) \setminus N(v^{\alpha,i})\big)$ is non-empty by the choice of $v^{\alpha,i}$ and the exclusion of neighbors of $p^{\alpha,i}$ in J_1.

Suppose that there exists u and w, which are incomparable in both orders. Then we can choose following vertices:

- $x_1 \in (N(u) \cap A \cap \mathcal{T}^\alpha) \setminus N(w, v^{\alpha,i})$
- $x_2 \in (N(w) \cap A \cap \mathcal{T}^\alpha) \setminus N(u, v^{\alpha,i})$
- $y_1 \in \big(N(u) \cap N(v^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha\big) \setminus N(w)$
- $y_2 \in \big(N(w) \cap N(v^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha\big) \setminus N(u)$

Then a path $x_1 - u - y_1 - v^{\alpha,i} - y_2 - w - x_2$ is an induced P_7, which is a contradiction. Thus, any pair of vertices of J_1 is comparable in at least one of the orders.

By Lemma 2.8 there exists a vertex $v_1^{\alpha,i}$ such that for any other vertex $w \in J_1$ we have $v_1^{\alpha,i} \leq_1 w$ or $v_1^{\alpha,i} \leq_2 w$. Let us choose vertices $p_1^{\alpha,i} \in \left(N(v_1^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha \cap N(v^{\alpha,i})\right)$ and $p_2^{\alpha,i} \in \left(N(v_1^{\alpha,i}) \cap A \cap \mathcal{T}^\alpha \setminus N(v^{\alpha,i})\right)$. Then for any $w \in J_1$ we have $w \in N(p_1^{\alpha,i}, p_2^{\alpha,i}) \cap N(X_B)$.

Consider now $v \in J \setminus N(p^{\alpha,i}, p_1^{\alpha,i}, p_2^{\alpha,i}, a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i})$; note that $v \notin J_1$. Let $u \in N(v) \cap \mathcal{T}^\alpha \cap A$. As $v \notin J_1$, $uv^{\alpha,i} \notin E(G)$. Let $Q \subseteq B$ be ap ath (maybe on just one vertex) such that $v - Q - v^{\alpha,i}$ is an induced path in G. Then, $c^{\alpha,i} - b^{\alpha,i} - a^{\alpha,i} - v^{\alpha,i} - Q - v - u$ contains an induced P_7 unless u is adjacent to at least one of the vertices $a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}$. This finishes the proof. ☐

For every $i \in [c]$ and $\alpha \in [d]$, we guess if $\overline{S_A^{\alpha,i}}$ is nonempty and, if this is the case, we guess the vertices $p^{\alpha,i}, p_1^{\alpha,i}, p_2^{\alpha,i}, a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}$ of Claim 3.5 and put

$$N(p^{\alpha,i}, p_1^{\alpha,i}, p_2^{\alpha,i}, a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}) \cap N(X_B)$$

into $\widetilde{S}$. Perform a symmetrical operation with the roles of A and B swapped.

Let $K_A = \{a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i} \mid i \in [c], \alpha \in [d]\}$. At this point, for every $w \in S_A \setminus \widetilde{S}$, all neighbors of w in $\mathcal{T} \cap A$ are in $N(K_A)$, while $N(K_A) \subseteq A \cup S$. Symmetrically we define K_B. Note that $|K_A|, |K_B| \leq 3cd$.

Let $A^1, A^2, \ldots, A^c$ and $B^1, B^2, \ldots, B^c$ be color classes of A and B, respectively. For every $i, j \in [c]$ and $\alpha \in [d]$, let $\overline{S_A^{\alpha,i,j}}$ be the set of those vertices of $\overline{S_A^{\alpha,i}} \setminus \widetilde{S}$ that have a neighbor in B^j. Symmetrically we define $\overline{S_B^{\alpha,i,j}}$.

Claim 3.6. *For each $i, j \in [c]$ and $\alpha \in [d]$, if $\overline{S_A^{\alpha,i,j}} \setminus \widetilde{S} \neq \emptyset$, then there exist $q^{\alpha,i,j} \in A$ and $v^{\alpha,i,j} \in \overline{S_A^{\alpha,i,j}}$ such that every vertex $v \in \overline{S_A^{\alpha,i,j}} \setminus \left(N(q^{\alpha,i,j}) \cap N(X_B)\right)$ satisfies $N(v) \cap B^j \subseteq N(v^{\alpha,i,j}) \cap B^j$.*

A symmetrical statement holds for $\overline{S_B^{\alpha,i,j}}$ for each $\alpha \in [d]$ and $i \in [c]$.

Proof. Fix $i, j \in [c]$ and $\alpha \in [d]$ and for brevity denote $J = \overline{S_A^{\alpha,i,j}}$.

Let us consider two orders on J:

- $\leq_1$: $u \leq_1 w \Leftrightarrow N(u) \cap A \cap \mathcal{T}^\alpha \subseteq N(w) \cap A \cap \mathcal{T}^\alpha$
- $\leq_2$: $u \leq_2 w \Leftrightarrow (N(u) \cap B^j) \supseteq (N(w) \cap B^j)$

Suppose that there exist vertices u and w in J, which are incomparable in both orders. Then we choose $x_1 \in N(u) \cap A \cap T^\alpha \setminus N(w)$ and $x_2 \in N(w) \cap A \cap T^\alpha \setminus N(u)$, and $y_1 \in N(u) \cap B^j \setminus N(w)$ and $y_2 \in N(w) \cap B^j \setminus N(u)$. Then $x_1 - u - y_1$ and $x_2 - w - y_2$ are induced paths on 3 vertices and there are no edges between these two paths. As $x_1, x_2 \in N(a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i})$, we can connect these two paths via vertices $a^{\alpha,i}, b^{\alpha,i}, c^{\alpha,i}$, finding then an induced P_7, which is a contradiction. Thus any pair of vertices is comparable in at least one of these orders.

By Lemma 2.8 there exists a vertex $v^{\alpha,i,j} \in J$ such that for any other vertex $w \in J$ we have $v^{\alpha,i,j} \leq_1 w$ or $v^{\alpha,i,j} \leq_2 w$. Let $q^{\alpha,i,j}$ be a vertex belonging to $N(v^{\alpha,i,j}) \cap A \cap \mathcal{T}^\alpha$. Then for each $w \in J$ we have $w \in N(q^{\alpha,i,j}) \cap N(X_B)$ or $N(w) \cap B^j \subseteq N(v^{\alpha,i,j}) \cap B^j$. This finishes the proof of the claim. ☐

For every $i, j \in [c]$ and $\alpha \in [d]$, guess if $\overline{S_A^{\alpha,i,j}}$ is nonempty and, if this is the case, guess the vertices $q^{\alpha,i,j}$ and $v^{\alpha,i,j}$ of Claim 3.6. Add $N(q^{\alpha,i,j}) \cap N(X_B)$ and $N(v^{\alpha,i,j}) \setminus N(K_A)$ to $\widetilde{S}$.

Note that in the last step we do not add any elements of $N(v^{\alpha,i,j}) \cap \mathcal{T} \cap A$ to $\widetilde{S}$. As $v^{\alpha,i,j} \notin \widetilde{S}$ prior to this step and $N(v^{\alpha,i,j}) \cap \mathcal{T} \cap B$ is of size at most d, we add at most d^2c^2 vertices of $\mathcal{T}$ to $\widetilde{S}$ in this step.

We perform a symmetric operation with A and B swapped.

At this moment, for every $v \in S_A \setminus \widetilde{S}$, we have $N(v) \cap A \cap \mathcal{T} \subseteq N(K_A)$ and $N(v) \cap B \subseteq \widetilde{S}$ and, symmetrically, for every $v \in S_B \setminus \widetilde{S}$, we have $N(v) \cap B \cap \mathcal{T} \subseteq N(K_B)$ and $N(v) \cap A \subseteq \widetilde{S}$.

Recall that $|X_B| \leq k$ and $|K_B| \leq 3cd$. Let $\widetilde{X_B}$ be the smallest connected subgraph of B, which contains $X_B \cup K_B$; as G is P_7-free, $|\widetilde{X_B}| \leq 7(k+3cd)$. Symmetrically define $\widetilde{X_A}$. Guess $\widetilde{X_A}$, $\widetilde{X_B}$ and add $N(\widetilde{X_A}) \cap N(\widetilde{X_B})$ to $\widetilde{S}$.

We now perform the following cleaning operation.

1. Guess all vertices of $\mathcal{T} \cap \widetilde{S}$ and, for every $v \in \mathcal{T} \cap \widetilde{S}$, guess all ancestors of v in $\mathcal{T}$. Add those ancestors to $\widetilde{S}$; this increases the size of $\mathcal{T} \cap \widetilde{S}$ by at most a factor of d. Now $\mathcal{T} \cap \widetilde{S}$ is a subtreedepth-d structure of $\mathcal{T}$.

2. Guess the depths and parent/child relation in $\mathcal{T}$ of vertices of $\mathcal{T} \cap \widetilde{S}$. In particular, this guess determines which pairs of vertices are incomparable in $\mathcal{T}$.

3. For every pair u, v of vertices of $\mathcal{T} \cap \widetilde{S}$ that are incomparable in $\mathcal{T}$, add $N(u) \cap N(v)$ to $\widetilde{S}$.

4. For every $v \in \mathcal{T}^d \cap \widetilde{S}$, add $N(v)$ to $\widetilde{S}$.

   Note that due to the first step, subsequent steps do not add any new vertex of $\mathcal{T}$ to $\widetilde{S}$.

After the above cleaning, for every $v \in V(G) \setminus \widetilde{S}$, all vertices of $N(v) \cap \mathcal{T} \cap \widetilde{S}$ are comparable in $\mathcal{T}$ and do not contain any vertex of $\mathcal{T}^d$. In particular, as $\mathcal{T}$ is maximal, v has a neighbor in $\mathcal{T} \setminus \widetilde{S}$. Consequently, every vertex of $S_A \setminus \widetilde{S}$ has a neighbor in $(A \setminus \widetilde{S}) \cap \mathcal{T}$ and every vertex of $S_B \setminus \widetilde{S}$ has a neighbor in $(B \setminus \widetilde{S}) \cap \mathcal{T}$.

We infer that the connected components of $G - \widetilde{S}$ are of the following types:

(clean) Contained in a connected component of $G - S$ (i.e., in A, in B, or in $G - (A \cup S \cup B)$), or

(dirty) Contained in $A \cup S \cup B$, with a nonempty intersection both with S and $A \cup B$.

Observe that the clean components are disjoint with $N(\widetilde{X_A})$ or with $N(\widetilde{X_B})$ (or both), while the dirty components have nonempty intersection both with $N(\widetilde{X_A})$ and $N(\widetilde{X_B})$, as $S_A \subseteq N(\widetilde{X_B})$, while every vertex of $S_A \setminus \widetilde{S}$ has a neighbor in $((A \setminus \widetilde{S}) \cap \mathcal{T})$, which in turn has a neighbor in $K_A \subseteq \widetilde{X_A}$, and symmetrically for B. Hence, the algorithm can distinguish clean and dirty components.

In what follows, in a number of steps we will add some vertices to $\widetilde{S}$, but we will be only adding vertices of S. After the updates, we will continue using the nomenclature of clean/dirty components of $G - \widetilde{S}$, and they always refer to the current value of $\widetilde{S}$. Note that, as long as we keep the invariant that every vertex of $S_A \setminus \widetilde{S}$ has a neighbor in $(A \setminus \widetilde{S}) \setminus \mathcal{T}$ (which is maintained trivially if we add only vertices of S to $\widetilde{S}$), then the method of distiguishing clean and dirty components from the previous paragraph still works.

We now branch if $S_A \setminus \widetilde{S} = \emptyset$. If this is the case, then every dirty component C satisfies $C \cap N(\widetilde{X_A}) = C \cap S$. We insert $C \cap N(\widetilde{X_A})$ into $\widetilde{S}$ for every dirty component C. As a result, $S \subseteq \widetilde{S}$ while $|\widetilde{S} \cap \mathcal{T}| = \mathcal{O}_{k,d}(1)$. We insert $(K := \widetilde{S}, \mathcal{D} := \emptyset, L := \emptyset)$ into $\mathcal{F}$ and conclude this branch.

Symmetrically we handle a branch $S_B \setminus \widetilde{S} = \emptyset$. In the remaining branch, we assume that both $S_A \setminus \widetilde{S}$ and $S_B \setminus \widetilde{S}$ are nonempty.

Let $\widehat{X_B} = \widetilde{X_B} \cap N(S_A \setminus \widetilde{S})$; note that as $S_A \setminus \widetilde{S} \neq \emptyset$ and $S_A \subseteq N(X_B)$, we have $\widehat{X_B} \neq \emptyset$. We symmetrically define $\widehat{X_A}$ and observe it is nonempty, too.

We now focus on connected components of $B \setminus \widetilde{S}$. Let C be such a component. Note that C has no neighbors in $S_A \setminus \widetilde{S}$; hence C is also a connected component of $G - \widetilde{S} - N(X_A)$, and thus can be guessed if needed.

Claim 3.7. *For every connected component C of $B \setminus \widetilde{S}$,*

1. *every $x \in \widehat{X_B}$ is either complete or anticomplete to C, and*

2. *if the whole $\widehat{X_B}$ is anticomplete to C, then the whole $\widetilde{X_B}$ is anticomplete to C.*

A symmetrical statement holds with A and B swapped.

Proof. For the first point, assume there exists $x \in \widehat{X_B}$ with an induced P_3 of the form $x - C - C$. Let y be any neighbor of x in $S_A \setminus \widetilde{S}$. Observe that there would exist an induced P_7 of the form $C - C - x - y - A - A - A$, a contradiction.

If C is antiadjacent to $\widehat{X_B}$ but has a neighbor in $\widetilde{X_B}$, then as $\widehat{X_B} \neq \emptyset$ there exists a path Q from $\widehat{X_B}$ via $\widetilde{X_B}$ to C of length at least 2; let x be its endpoint in $\widehat{X_B}$ and let y be a neighbor of x in $S_A \setminus \widetilde{S}$. Then there exists a P_7 of the form $Q - y - A - A - A$, a contradiction. $\square$

Claim 3.7 distinguishes two types of components of $B \setminus \widetilde{S}$: those antiadjacent to $\widetilde{X_B}$ and those complete to at least one vertex of $\widehat{X_B}$. Let $\mathcal{B}_0$ and $\mathcal{B}_1$ be the families of components of the first and second type, respectively. Symmetrically define $\mathcal{A}_0$ and $\mathcal{A}_1$.

Recall that every vertex of $S_B \setminus \widetilde{S}$ has a neighbor in $(B \setminus \widetilde{S}) \cap \mathcal{T}$, which in turn has a neighbor in $K_B \subseteq \widetilde{X_B}$. This implies the following.

Claim 3.8. *Every vertex of $S_B \setminus \widetilde{S}$ has a neighbor in a component of $\mathcal{B}_1$. Symmetrically, every vertex of $S_A \setminus \widetilde{S}$ has a neighbor in a component of $\mathcal{A}_1$.*

For $v \in S_B \setminus \widetilde{S}$, let $N_{\mathcal{B}_0}(v) = \{C \in \mathcal{B}_0 \mid C \cap N(v) \neq \emptyset\}$ and $N_{\mathcal{B}_1}(v) = \{C \in \mathcal{B}_1 \mid C \cap N(v) \neq \emptyset\}$.

We now sort out adjacencies between $S_B \setminus \widetilde{S}$ and $\mathcal{B}_0$. Let $S_{B,0}$ be the set of vertices belonging to $S_B \setminus \widetilde{S}$ which have a neighbor in a component of $\mathcal{B}_0$. Let $S^1_{B,0}, S^2_{B,0}, \ldots, S^c_{B,0}$ be color classes of $S_{B,0}$. Symmetrically, we define $S_{A,0}$ and color classes $S^1_{A,0}, \ldots, S^c_{A,0}$.

Claim 3.9. *For every $i \in [c]$, if $S^i_{B,0}$ is nonempty, then there exist two components $B_{0,i,0}$ and $B_{0,i,1}$ of $B \setminus \widetilde{S}$ such that $S^i_{B,0} \subseteq N(B_{0,i,0} \cup B_{0,i,1})$.*

A symmetrical statement holds with A and B swapped.

Proof. Fix $i \in [c]$. Then we consider two orders on $S^i_{B,0}$:

- $\leq_1$: $x \leq_1 y \Leftrightarrow N_{\mathcal{B}_1}(x) \subseteq N_{\mathcal{B}_1}(y)$;
- $\leq_2$: $x \leq_2 y \Leftrightarrow N_{\mathcal{B}_0}(x) \subseteq N_{\mathcal{B}_0}(y)$;

Suppose that there exist x and y, which are incomparable in both orders. Then we can choose

- $C_x \in N_{\mathcal{B}_1}(x) \setminus N_{\mathcal{B}_1}(y)$,
- $C_y \in N_{\mathcal{B}_1}(y) \setminus N_{\mathcal{B}_1}(x)$,
- $D_x \in N_{\mathcal{B}_0}(x) \setminus N_{\mathcal{B}_0}(y)$,
- $D_y \in N_{\mathcal{B}_0}(y) \setminus N_{\mathcal{B}_0}(x)$.

There are no edges between $C_x \cup D_x \cup \{x\}$ and $C_y \cup D_y \cup \{y\}$. Then, there is an induced P_7 of the form $D_x - x - C_x - \widetilde{X_B} - C_y - y - D_y$, a contradiction. Thus any pair of vertices is comparable in at least one order.

By Lemma 2.8 we can choose a vertex $v^i \in S^i_{B,0}$ such that for any other vertex $y \in S^i_{B,0}$ we have $v^i \leq_1 y$ or $v^i \leq_2 y$. Hence, any $B_{0,i,0} \in N_{\mathcal{B}_0}(v^i)$ (which exists by the definition of $S_{B,0}$) and any $B_{0,i,1} \in N_{\mathcal{B}_1}(v^i)$ (which we exists by Claim 3.8) would do the job. $\square$

As discussed, components of $\mathcal{B}_0 \cup \mathcal{B}_1$ are components of $G - \widetilde{S} - N(X_A)$ and can be guessed. For every $i \in [c]$, we guess $B_{0,i,0}$ and $B_{0,i,1}$ and add $N(B_{0,i,0} \cup B_{0,i,1}) \cap N(X_A)$ to $\widetilde{S}$.

We perform a symmetrical operation for A. As a result, no vertex of $S_B \setminus \widetilde{S}$ has a neighbor in a component of $\mathcal{B}_0$ and no vertex of $S_A \setminus \widetilde{S}$ has a neighbor in $\mathcal{A}_0$. Also, we added to $\widetilde{S}$ only vertices of S, so in particular no vertex of $\mathcal{T}$ was added to $\widetilde{S}$ and the families $\mathcal{B}_0$, $\mathcal{B}_1$, $\mathcal{A}_0$, and $\mathcal{A}_1$ stay intact.

We say that $x \in S_B \setminus \widetilde{S}$ is *mixed* to a component $C \in \mathcal{B}_1$ if there exists an induced path $x - C - C$ or, equivalently, x is neither complete nor anticomplete to C. A component $C \in \mathcal{B}_1$ is *problematic* if there is $x \in S_B \setminus \widetilde{S}$ that is mixed to C. For $x \in S_B \setminus \widetilde{S}$ we denote by $\texttt{Prob}(x) \subseteq \mathcal{B}_1$ the set of components x is mixed to and by $\texttt{All}(x) \subseteq \mathcal{B}_1$ the set of components x is mixed or complete to. Let S_B^P be the set of all vertices of $S_B \setminus \widetilde{S}$ that are mixed to at least one component of $\mathcal{B}_1$ (i.e., $S_B^P = \{x \in S_B \setminus \widetilde{S} \mid \texttt{Prob}(x) \neq \emptyset\}$) and let $S_B^{P,1}, S_B^{P,2}, \ldots, S_B^{P,c}$ be color classes of S_B^P. We introduce symmetrical definitions with A and B swapped.

Claim 3.10. *For each $i \in [c]$ and any pair of vertices $v_1, v_2 \in S_B^{P,i}$ we have either $\texttt{Prob}(v_1) \subseteq \texttt{All}(v_2)$ or $\texttt{Prob}(v_2) \subseteq \texttt{All}(v_1)$. A symmetrical statement holds for any pair of vertices of $S_A^{P,i}$ for each $i \in [c]$.*

Proof. Fix $i \in [c]$. Suppose that there exist vertices v_1 and v_2 in $S_B^{P,i}$ and components $C_{v_1} \in \texttt{Prob}(v_1) \setminus \texttt{All}(v_2)$ and $C_{v_2} \in \texttt{Prob}(v_2) \setminus \texttt{All}(v_1)$. Then there exist two induced P_3s $v_1 - C_{v_1} - C_{v_1}$ and $v_2 - C_{v_2} - C_{v_2}$, which can be connected via $\widetilde{X_A}$, so we can find a P_7, a contradiction. $\square$

Claim 3.11. *For every $i \in [c]$ there exists components $C_{v^i}, C^1_{u^i}, C^2_{u^i} \in \mathcal{B}_1$ such that $S_B^{P,i} \subseteq \big(N(C_{v^i}) \cup N(C^1_{u^i}) \cup N(C^2_{u^i})\big) \cap N(\widehat{X_A})$. A symmetrical statement holds for $S_A^{P,i}$ for each $i \in [c]$.*

Proof. Fix $i \in [c]$. Let $v^i \in S_B^{P,i}$ be a vertex, which has neighbors in the smallest number of problematic components. Choose any $C_{v^i} \in \texttt{Prob}(v^i)$. Let w be any vertex in $S_B^{P,i} \setminus (N(C_{v^i}) \cap N(X_A))$. Since $C_{v^i} \in \texttt{Prob}(v^i) \setminus \texttt{All}(w)$, by Claim 3.10 we know that $\texttt{Prob}(w) \subseteq \texttt{All}(v^i)$. By the minimality of v^i, we know that $\texttt{All}(w) \setminus \texttt{All}(v^i) \neq \emptyset$. Let us consider two orders on $S_B^{P,i} \setminus (N(C_{v^i}) \cap N(X_A))$:

- $\leq_1$: $u \leq_1 w \Leftrightarrow \{C \in \texttt{All}(v^i) \mid N(u) \cap C \neq \emptyset\} \subseteq \{C \in \texttt{All}(v^i) \mid N(w) \cap C \neq \emptyset\}$
- $\leq_2$: $u \leq_1 w \Leftrightarrow \{C \in \mathcal{B}_1 \setminus \texttt{All}(v^i) \mid N(u) \cap C \neq \emptyset\} \subseteq \{C \in \mathcal{B}_1 \setminus \texttt{All}(v^i) \mid N(w) \cap C \neq \emptyset\}$

Note that for every $u \in S_B^{P,i} \setminus (N(C_{v^i}) \cap N(\widehat{X_A}))$ the set $\{C \in \texttt{All}(v^i) \mid N(u) \cap C \neq \emptyset\}$ is non-empty, as by Claim 3.10 we have $\emptyset \neq \texttt{Prob}(u) \subseteq \texttt{All}(v^i)$. Similarly, for every $u \in S_B^{P,i} \setminus (N(C_{v^i}) \cap N(\widehat{X_A}))$ the set $\{C \in \mathcal{B}_1 \setminus \texttt{All}(v^i) \mid N(u) \cap C \neq \emptyset\}$ is non-empty by the choice of v^i and the exclusion of $N(C_{v^i})$.

Suppose that there exist u and w which are incomparable in both these orders. Then we can choose the following components:

- $C^1_u \in \texttt{All}(v^i)$ such that $N(u) \cap C^1_u \neq \emptyset$ and $N(w) \cap C^1_u = \emptyset$
- $C^2_u \in \mathcal{B}_1 \setminus \texttt{All}(v^i)$ such that $N(u) \cap C^2_u \neq \emptyset$ and $N(w) \cap C^2_u = \emptyset$
- $C^1_w \in \texttt{All}(v^i)$ such that $N(w) \cap C^1_w \neq \emptyset$ and $N(u) \cap C^1_u = \emptyset$
- $C^2_w \in \mathcal{B}_1 \setminus \texttt{All}(v^i)$ such that $N(w) \cap C^2_w \neq \emptyset$ and $N(u) \cap C^1_u = \emptyset$

Then there exists a path $C^2_u - u - C^1_u - v^i - C^1_w - w - C^2_w$ that contains a P_7, a contradiction. Hence, any pair of vertices is comparable in at least one order.

Therefore by Lemma 2.8 there exists a vertex u^i such that for any other vertex $w \in S_B^{P,i} \setminus (N(C_v) \cap N(\widehat{X_A}))$ we have $u^i \leq_1 w$ or $u^i \leq_2 w$. Let $C^1_{u^i} \in \texttt{All}(v^i)$ and $C^2_{u^i} \in \mathcal{B}_1 \setminus \texttt{All}(v^i)$ such that $N(u^i) \cap C^1_u \neq \emptyset$ and $N(u^i) \cap C^2_u \neq \emptyset$. Then by the choice of u^i we have that $S_B^{P,i} \setminus (N(C_{v^i}) \cap N(\widehat{X_A})) \subseteq (N(C^1_{u^i}) \cup N(C^2_{u^i})) \cap N(\widehat{X_A})$. Therefore $S_B^{P,i} \subseteq \big(N(C_{v^i}) \cup N(C^1_{u^i}) \cup N(C^2_{u^i})\big) \cap N(\widehat{X_A})$.

By symmetry, we show that there exist sought components $C_{v^i}, C^1_{u^i}, C^2_{u^i} \in \mathcal{A}_1$ for each $S_A^{P,i}$. $\square$

For every $i \in [c]$, we guess components C_{v^i}, $C^1_{u^i}$ and $C^2_{u^i}$ in $\mathcal{B}_1$ whose existence is guaranteed by Claim 3.11, and add $\big(N(C_{v^i}) \cup N(C^1_{u^i}) \cup N(C^2_{u^i})\big) \cap N(\widehat{X_A})$ to $\widetilde{S}$. We perform symmetric operation for $S_A^{P,i}$.

By Claim 3.11, we have $S_A^P \cup S_B^P \subseteq \widetilde{S}$, while we added only vertices of S to $\widetilde{S}$. Now, every vertex of $S_B \setminus \widetilde{S}$ is complete or anticomplete to every component of $\mathcal{B}_1$ (and anticomplete to every component of $\mathcal{B}_0$).

Consequently, for every dirty component C, every connected component of $C \cap B$ is a module of $G[C]$ and every connected component of $C \cap A$ is a module of $G[C]$. Furthermore, observe that thanks to Claim 3.7 for every dirty component C, we have $C \cap N(\widetilde{X_A}) = (C \cap A) \cup (C \cap S_B)$ and $C \cap N(\widetilde{X_B}) = (C \cap B) \cup (C \cap S_A)$. That is, $(C \cap N(\widetilde{X_A}))$ and $(C \cap N(\widetilde{X_B}))$ is a partition of C which the algorithm can compute. Moreover, every connected component of $G[C \cap N(\widetilde{X_A})]$ is either a component of $A \setminus \widetilde{S}$ or a component of $S_B \setminus \widetilde{S}$ and every connected component of $G[C \cap N(\widetilde{X_B})]$ is either a component of $B \setminus \widetilde{S}$ or a component of $S_A \setminus \widetilde{S}$.

Since every connected component of $C \cap B$ and of $C \cap A$ is a module of $G[C]$, we know that all components of $G[C \cap N(\widetilde{X_A})]$ or of $G[C \cap N(\widetilde{X_B})]$ that are *not* modules of $G[C]$ are part of $\widetilde{S}$. That is, we perform the following step exhaustively: while there exists a dirty component C and a component C' of either $G[C \cap N(\widetilde{X_A})]$ or $G[C \cap N(\widetilde{X_B})]$ that is not a module of $G[C]$, move C' to $\widetilde{S}$. This step, again, adds only vertices belonging to S to $\widetilde{S}$.

In the end, for every dirty component C, $(C \cap \widetilde{X_A}, C \cap \widetilde{X_B})$ is a partition of C such that every component of $G[C \cap \widetilde{X_A}]$ and every component of $G[C \cap \widetilde{X_B}]$ is a module of $G[C]$. Let $\mathcal{D}$ be the set of dirty components and, for $C \in \mathcal{D}$, let $L(C) = C \cap \widetilde{X_A}$. Then, $(K := \widetilde{S}, \mathcal{D}, L)$ satisfies the requirements of the lemma for (A, S, B) and we add this tuple to $\mathcal{F}$.

This finishes the proof of Theorem 3.1. ☐

4 P_7-free bipartite graphs

In this section we focus on P_7-free bipartite graphs. We prove the following theorem.

Theorem 4.1. *There exists an algorithm that, given a P_7-free bipartite graph G and an integer $d > 0$, runs in time $n^{2^{2^{\mathcal{O}(d^3)}}}$ and computes a family $\mathcal{C}$ of subsets of $V(G)$ with the following guarantee: for every treedepth-d structure $\mathcal{T}$ in G, there exists a tree decomposition (T, β) of G such that for every $t \in V(T)$ we have $\beta(t) \in \mathcal{C}$ and $|\beta(t) \cap \mathcal{T}| = 2^{2^{\mathcal{O}(d^3)}}$.*

The general approach is to reduce to the case of *chordal bipartite graphs.* Recall that a bipartite graph is *chordal bipartite* if its every cycle of length longer than 4 has a chord (i.e., the only induced cycles are of length 4).

The class of MWIS problems is tractable on chordal bipartite graphs thanks to the following result of Kloks, Liu, and Poon (and the general algorithm of Fomin, Todinca, and Villanger [19]).

Theorem 4.2 (Corollary 2 of [29])**.** *A chordal bipartite graph on n vertices and m edges has $\mathcal{O}(n + m)$ minimal separators.*

Observe that P_7-free bipartite graphs are "almost" chordal bipartite: they only additionally allow C_6 as an induced subgraph. Our approach is to add edges to the input graph so that it becomes chordal bipartite, without destroying the sought solution. To this end, the following folklore characterization will be handy.

Lemma 4.3 (folklore)**.** *Let G be a bipartite graph with bipartition V_1, V_2. Then, G is chordal bipartite if and only if for every minimal separator S of G, $S \cap V_1$ is complete to $S \cap V_2$ (i.e., the separator induces a complete bipartite graph, also called a* biclique*).*

Proof. In one direction, assume that G contains a cycle C of length $\ell \geq 6$ as an induced subgraph and let $x_1, x_2, \ldots, x_\ell$ be consecutive vertices of C. Then, $\{x_2, x_3\}$ and $\{x_5, \ldots, x_\ell\}$ are anticomplete. Furthermore, any minimal separator separating these sets contains x_1 and x_4, which are nonadjacent and on the opposite bipartition sides of G

In the other direction, let S be a minimal separator of G with two full components A, B and vertices $x \in S \cap V_1$, $y \in S \cap V_2$, $xy \notin E(G)$. Observe that, thanks to the bipartiteness of G, a shortest path from x to y via A is of length at least 3, and similarly via B. These two shortest path together form an induced cycle of length at least 6. $\square$

Lemma 4.3 motivates the following process of completing a P_7-free bipartite graph G into a chordal bipartite graph: while G contains a minimal separator S that violates the statement of Lemma 4.3, complete $G[S]$ into a biclique. In Section 4.1 we analyse this process and show that it is well-behaved on P_7-free graphs, in particular, does not lead outside the class of P_7-free bipartite graphs.

4.1 Completing to a chordal bipartite graph

Throughout the rest of this section we assume that the input graph G has a fixed bipartition into sets V_1, V_2 (this is an ordered partition). We remark that we will often look at certain induced subgraphs of G that are not necessarily connected, but a vertex never changes its side of the bipartition.

Lemma 4.3 motivates the following definition. Let G be a bipartite graph with bipartition V_1, V_2, and let S be a minimal separator of G. We say that S *induces a biclique* if $S \cap V_1$ is complete to $S \cap V_2$. The operation of *completing S into a biclique* turns G into a graph $G + F := (V(G), E(G) \cup F)$, where $F = \{uv \mid u \in S \cap V_1, v \in S \cap V_2, uv \notin E(G)\}$.

The following lemma is pivotal to our completion process.

Lemma 4.4. *Let G be a P_7-free bipartite graph. Let S be a minimal separator and let A and B be two full components of S. Let $G + F$ be the result of completing S into a biclique. Then $G + F$ is also P_7-free.*

Proof. By contradiction, suppose that $G + F$ contains a P_7. Let Q be a P_7 in $G + F$, which minimizes $|E(Q) \cap F|$ among all P_7s contained in $G + F$.

As G is P_7-free, $E(Q) \cap F \neq \emptyset$. Consequently, $|V(Q) \cap S| \geq 2$. We consider several cases.

Case A. $|V(Q) \cap S| = 2$.
Let $V(Q) \cap S = \{x, y\}$. Note that $E(Q) \cap F = \{xy\}$, i.e., $xy \in F$ and $xy \notin E(G)$. Let Q_x and Q_y be the components of $Q - \{xy\}$ that contain x and y, respectively.

Since $|V(Q) \setminus S| = 5$, without loss of generality assume that Q_x is of length at most 2 and Q_y is of length at least 3. Note that $Q_y \setminus \{y\}$ is contained in a single component of $G - S$; without loss of generality assume that this component is not A. Let R be a shortest path from x to y via A; note that R has at least 3 edges. Then, the concatenation of R and Q_y is an induced path in G on at least 7 vertices, a contradiction.

Case B. $|V(Q) \cap S| > 2$.
As S induces a biclique in $G + F$, we have $|V(Q) \cap S| = 3$ and $V(Q) \cap S$ consists of three consecutive vertices of Q; call these vertices x, y, z (in the order of appearance on Q, i.e., $xy, yz \in E(G) \cup F$).

As $E(Q) \cap F \neq \emptyset$, either $xy \in F$ or $yz \in F$ (or both). Without loss of generality, we can assume that $xy \in F$. Note that yz may or may not belong to F.

Let Q_x and Q_z be the components of $Q - \{y\}$ that contain x and z, respectively. Each of $Q_x - \{x\}$ and $Q_z - \{z\}$ is contained in a single component of $G - S$. By the symmetry between A and B, we assume that $Q_z - \{z\}$ is not contained in A.

Let R be a shortest path from x to y via A; as before, R is of length at least 3. Let p be the neighbor of x on R.

In the following we consider cases depending on the position of x on Q.

1. Vertex x is the first or the second vertex of Q, i.e., Q_x is of length at most 1.
   If $pz \in E(G)$, then substituting y with p on Q yields a P_7 in G. If $pz \notin E(G)$, then the concatenation of Q_z, the edge yz, and $R - \{x\}$ is an induced path on at least 7 vertices either in G (if $yz \in E(G)$) or in $G + F$ with strictly fewer edges of F than Q (if $yz \in F$). In all cases, we get a contradiction.

2. There are at least three vertices before x on the path Q, i.e., Q_x is of length at least 3.
   If $Q_x - \{x\}$ is contained in A, then let R' be a shortest path from x to y via B, and otherwise let $R' := R$. Then, the concatenation of R' and Q_x is an induced path in G on at least 7 vertices, a contradiction.

3. Vertices x, y, z are the middle vertices of Q, i.e., both Q_x and Q_z are of length 2.
   If there exists a component $C \in \mathrm{cc}(G - S)$ such that $x, z \in N(C)$ and $V(Q) \cap C = \emptyset$, then we can connect x and z via a shortest path R_C in C. Then the concatenation of Q_x, R_C, and Q_z is an induced path on at least 7 vertices, a contradiction.

   If there is no such component C, then Q must be of a form $a_1 - a_2 - x - y - z - b_2 - b_1$, where $a_1, a_2 \in A$ and $b_1, b_2 \in B$. Note that if there is a path connecting x and z via A or B with at least 3 intermediate vertices, then we can extend this path with either a_2 and a_1 or b_2 and b_1 to get a P_7. Similarly, if there a path connecting x and y (or y and z if $yz \in F$) via A or B with at least four intermediate vertices, then we can also extend it with a_2 and a_1 or b_2 and b_1 to get a P_7. Consequently, R is of the form $x - p - q - y$ with $p, q \in A$.

   We make the following observation. If there exists $u \in A$ such that $x, z \in N(u)$ and $a_1 \notin N(u)$, then a path $a_1 - a_2 - x - u - z - b_2 - b_1$ is a P_7 in G. Thus any vertex in A which is a neighbor of x and z is also a neighbor of a_1. By symmetry any vertex in B which is a neighbor of x and z is also a neighbor of b_1.

   Suppose now that $pz \in E(G)$. By the observation above, we have $pa_1 \in E(G)$. If $a_2 q \notin E(G)$, then a path $a_2 - a_1 - p - q - y - B - B$ is a P_7 in G, a contradiction (here, $y - B - B$ stands for any two-edge path from y into B, which exists as B is connected and contains at least 2 vertices). Otherwise, a path $a_1 - a_2 - q - y - z - b_2 - b_1$ is an induced P_7 in $G + F$, which has strictly fewer edges in F than Q, a contradiction.

   Therefore, $pz \notin E(G)$. If $yz \in E(G)$, then a path $x - p - q - y - z - b_2 - b_1$ is an induced P_7 in G. Thus $yz \in F$. Let R' be the shortest path connecting y and z via B. We have that $R' = z - p' - q' - y$ for some $p', q' \in B$. If $p'x \in E(G)$, then we can conclude in an analogous way to a case if $pz \in E(G)$. Therefore we can assume that $p'x \notin E(G)$. The path $x - p - q - y - q' - p' - z$ is an induced P_7 in G then, which is the final contradiction.

This completes the proof. ☐

In the next lemma, we establish that completing a minimal separator into a biclique in a P_7-free graph does not create new C_6s.

Lemma 4.5. *Let G be a P_7-free bipartite graph. Let S be a minimal separator in G and let A and B two full components of S. Let $G + F$ be the result of completing S into a biclique. Let C be an induced C_6 in $G + F$. Then, C is also an induced C_6 in G.*

Proof. By contradiction, suppose that $E(C) \cap F \neq \emptyset$. As S in a biclique in $G + F$, $V(C) \cap S$ contains either two or three consecutive vertices of C. Thus, $P := C \setminus S$ is a path and belongs to exactly one component of $G - S$. Without loss of generality, we can assume that P does not lie in B, i.e., $C \cap B = \emptyset$.

Let $x, z \in V(C) \cap S$ be the two vertices of C adjacent on C to the vertices of P. Note that $xz \notin E(G)$: either $|V(C) \cap S| = 3$ and x, z are on the same biparteness side of G or $|V(C) \cap S| = 2$ and then $xz \in F$.

Let R be the shortest path from x to z via B. Then, $R \cup P$ induce a hole C' in G. Since R is of length at least 2, C' is not shorter than C. Since G is P_7-free, C' is a six-vertex hole. This can only happen if R is of length 2 and $V(C) \cap S = \{x, y, z\}$ for some $y \in S$. Then, $C = x - y - z - r - q - p - x$, where p, q, r lie in a single component of $G - S$ different than B. Without loss of generality, we can assume that $yz \in F$. Then we can find a shortest path R', which connects y and z via B; it is of length at least 3. Then the path $R - r - q - p$ is an induced path in G on at least 7 vertices, a contradiction. ☐

Let G be a P_7-free bipartite graph with fixed bipartition V_1, V_2, and let $\mathcal{T}$ be a treedepth-d structure in G. A tuple (C, x, y) is a *bad* C_6 (with respect to $\mathcal{T}$) in G if C is an induced six-vertex cycle in G, and $x \in V_1$, $y \in V_2$ are two vertices that are at the same time (a) opposite vertices of C, and (b) incomparable vertices of $\mathcal{T}$. That is, if one adds the edge xy to G (which can be done without violating the biparteness of G), then one breaks the treedepth-d structure $\mathcal{T}$, i.e., $\mathcal{T}$ is not a treedepth-d structure in $G + xy$.

We have the following simple corollary of Lemma 4.5.

Lemma 4.6. *Let G be a P_7-free bipartite graph, let $\mathcal{T}$ be a treedepth-d structure in G, let S be a minimal separator in G, and let $G + F$ be a result of completing S into a biclique. Assume that there is no bad C_6 in G with respect to $\mathcal{T}$. Then, $\mathcal{T}$ is a treedepth-d structure in $G + F$ and, furthermore, there is no bad C_6 in $G + F$ with respect to $\mathcal{T}$.*

Proof. Let A and B be two full components of S. Let $xy \in F$ be arbitrary. Let R_A and R_B be shortest paths from x to y via A and B, respectively. Then, the concatenation of R_A and R_B is an induced cycle C in G; since G is P_7-free, C is a six-vertex cycle with x and y being two opposite vertices. Since (C, x, y) is not a bad C_6 w.r.t. $\mathcal{T}$, the addition of the edge xy does not break the treedepth-d structure $\mathcal{T}$. Since the choice of $xy \in F$ was arbitrary, $\mathcal{T}$ is a treedepth-d structure in $G + F$. Furthermore, Lemma 4.5 ensures that every C_6 in $G + F$ is also present in G; thus, $G + F$ does not contain a bad C_6 w.r.t. $\mathcal{T}$. $\square$

We conclude this section with the following enumeration.

Lemma 4.7. *There exists an algorithm that, given a P_7-free bipartite graph G and an integer d, runs in polynomial time and returns a family $\mathcal{C}$ of subsets of $V(G)$ of size $\mathcal{O}(|V(G)|^5)$ with the following property: for every treedepth-d structure $\mathcal{T}$ in G that admits no bad C_6, there exists a tree decomposition (T, β) of G such that for every $t \in V(T)$, the set $\beta(t)$ belongs to $\mathcal{C}$ and $\beta(t)$ contains at most d elements of $\mathcal{T}$.*

Proof. Consider the following process. Start with $\widehat{G} := G$. While $\widehat{G}$ is not chordal bipartite, find an induced six-vertex cycle C in G, fix two opposite vertices x and y in C, find a minimal separator S in G that contains x and y (note that any minimal separator separating the two components of $C - \{x, y\}$ would do), complete S into a biclique obtaining $\widehat{G} + F$, and set $\widehat{G} := \widehat{G} + F$.

Lemma 4.4 ensures that $\widehat{G}$ stays P_7-free bipartite. Lemma 4.6 ensures that every treedepth-d structure $\mathcal{T}$ in G that admits no bad C_6 remains so in $\widehat{G}$.

By Theorem 4.2, $\widehat{G}$ has $\mathcal{O}(|V(G)|^2)$ minimal separators. By Theorem 2.2, $\widehat{G}$ has $\mathcal{O}(|V(G)|^5)$ potential maximal cliques and they can be enumerated in polynomial time. We return the list of all potential maximal cliques of $\widehat{G}$ as $\mathcal{C}$.

As shown in [13], there exists a minimal chordal completion F of $\widehat{G}$ such that for every maximal clique Ω of $\widehat{G} + F$, the set $\Omega \cap \mathcal{T}$ is contained in a single leaf-to-root path in $\mathcal{T}$ and thus is of size at most d. Since all these maximal cliques are enumerated in $\mathcal{C}$, any clique tree of $\widehat{G} + F$ serves as the promised tree decomposition (T, β). $\square$

4.2 Cleaning

In this section we define a branching step that cleans a specific part of the graph. The step will be general enough to be applicable in many contexts.

Let G be a P_7-free bipartite graph. We say that a triple (A, B, C) of pairwise disjoint vertex sets of G is $\mathbb{V}$*-free* if there are no two anticomplete P_3s of the form $A - B - C$. Note that such a configuration naturally appears in a P_7-free bipartite graph if $A \cup C$ is in one side on the bipartition, B is in the other side, and there is an additional vertex v in the same side as B such that $A \subseteq N(v)$ but $C \cap N(v) = \emptyset$.

In our case, $\mathbb{V}$-free sets appear naturally in the following context.

Lemma 4.8. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2. Let A and C be two disjoint subsets contained in V_i, for some $i \in \{1,2\}$, and let $B \subseteq V_{3-i}$. Furthermore, assume that there exists a vertex $v \in V_{3-i} \setminus B$ such that $A \subseteq N(v)$ but $C \cap N(v) = \emptyset$. Then, (A,B,C) is $\mathbb{W}$-free.*

Proof. Any two anticomplete P_3s of the form $A - B - C$, together with v, induce a P_7 in G. $\square$

Recall that in our setting, we have some unknown treedepth-d structure $\mathcal{T}$ in G and we are building a container for it. That is, we are happy with inserting any number of vertices of G into the container, as long as we are guaranteed that we insert only a constant number of vertices of $\mathcal{T}$ along the way. In the case of a $\mathbb{W}$-free triple (A,B,C), we would like to simplify this part of G by filtering out vertices of B that have neighbors both in A and in C. To achieve this goal, we will guess a set $X \subseteq A \cup B \cup C$ that on one hand contains (in one of the branches) at most a constant number of vertices of the fixed unknown $\mathcal{T}$, and on the other hand satisfies the following: no $b \in B \setminus X$ has a neighbor both in $A \setminus X$ and in $C \setminus X$.

To this end, we will rely on the following simple yet powerful observation. Observe that if (A,B,C) is $\mathbb{W}$-free and both $A \cup C$ and B are independent sets (which happen, in particular, if $A \cup C$ is on one side of the bipartition and B is on the other side of the bipartition), then, for every distinct $b_1, b_2 \in B$ either $N(b_1) \cap A$ and $N(b_2) \cap A$ are comparable by inclusion or $N(b_1) \cap C$ and $N(b_2) \cap C$ are comparable by inclusion. This allows to use Lemma 2.8 on subsets of B, with the orders $\leq_1$ and $\leq_2$ being inclusion of the neighborhoods in A and C, respectively. This observation is the engine of the following lemma that formalizes our goal of filtering out vertices of B that have neighbors both in A and in C.

Lemma 4.9. *Fix an integer d. Let G be a graph, and let (A,B,C) be a $\mathbb{W}$-free triple in G with both $A \cup C$ and B being independent set. Then one can in polynomial time enumerate a family $\mathcal{F}$ of subsets of $A \cup B \cup C$ of size at most $n^{4(d+1)^2}$ with the following guarantee:*

- *For every $X \in \mathcal{F}$, there is no $b \in B \setminus X$ with both $N(b) \cap (A \setminus X)$ and $N(b) \cap (C \setminus X)$ nonempty.*
- *For every treedepth-d structure $\mathcal{T}$ in G, there exists $X \in \mathcal{F}$ such that $|X \cap \mathcal{T}| \leq d^2(d+1)$.*

Proof. Fix a treedepth-d structure $\mathcal{T}$ in G. We will describe the process of enumerating the elements of $\mathcal{F}$ as a branching algorithm, guessing some properties of $\mathcal{T}$. The number of leaves of the branching will be polynomial in the size of G and in each leaf of the branching we will output one set X that satisfies the second property for every $\mathcal{T}$ that agrees with the guesses made in this leaf.

For every $b \in B$, let $\iota_A(b)$ be the maximum integer $1 \leq \alpha \leq d$ such that $N(b) \cap A \cap \mathcal{T}^\alpha$ contains at least two vertices; $\iota_A(b) = 0$ if such a α does not exist. Similarly define $\iota_C(b)$ with respect to $N(b) \cap C \cap \mathcal{T}^\alpha$. For $0 \leq \alpha, \beta \leq d$, let $B_{\alpha,\beta} = \{b \in B \mid \iota_A(b) = \alpha \wedge \iota_C(b) = \beta\}$. Note that sets $B_{\alpha,\beta}$ form a partition of B.

Initialize $X = \emptyset$. For every $0 \leq \alpha, \beta \leq d$, we we will guess some set of vertices and include them into X. We will argue that there will be a branch where the vertices guessed for α, β ensure that every vertex from $B_{\alpha,\beta} \setminus X$ satisfies the first statement of the lemma. Thus, for (at least) one of sets X generated in the process, the statement will hold for every vertex in $B \setminus X$.

Consider fixed $0 \leq \alpha, \beta \leq d$. If $B_{\alpha,\beta}$ is empty, there is nothing to do, so assume otherwise. Define the following two orders $\leq_1$ and $\leq_2$ on $B_{\alpha,\beta}$:

$$\begin{aligned}
b \leq_1 b' &\iff N(b) \cap A \supseteq N(b') \cap A, && \text{if } \alpha = 0,\\
b \leq_1 b' &\iff N(b) \cap A \cap \mathcal{T}^\alpha \subseteq N(b') \cap A \cap \mathcal{T}^\alpha && \text{if } \alpha > 0,\\
b \leq_2 b' &\iff N(b) \cap C \supseteq N(b') \cap C, && \text{if } \beta = 0,\\
b \leq_2 b' &\iff N(b) \cap C \cap \mathcal{T}^\beta \subseteq N(b') \cap C \cap \mathcal{T}^\beta && \text{if } \beta > 0.
\end{aligned}$$

Apply Lemma 2.8 to $B_{\alpha,\beta}$ with $\leq_1$ and $\leq_2$, obtaining a vertex $b^*_{\alpha,\beta}$.

If $\alpha = 0$, guess $b^*_{\alpha,\beta}$ and add $N(b^*_{\alpha,\beta}) \cap A$ to X. This adds at most d vertices of $\mathcal{T}$ to X, while adds to X all vertices of $N(b) \cap A$ such that $b \in B_{\alpha,\beta}$ and $b^*_{\alpha,\beta} \leq_1 b$.

If $\alpha > 0$, guess two elements $a^1_{\alpha,\beta}, a^2_{\alpha,\beta} \in N(b^*_{\alpha,\beta}) \cap A \cap \mathcal{T}^\alpha$ and add $N(a^1_{\alpha,\beta}) \cap N(a^2_{\alpha,\beta}) \cap B$ to X. This adds at most $(\alpha - 1)$ vertices of $\mathcal{T}$ to X, while adds to X every $b \in B_{\alpha,\beta}$ such that $b^*_{\alpha,\beta} \leq_1 b$.

Perform symmetrical operation for β and C: If $\beta = 0$, guess $b^*_{\alpha,\beta}$ and add $N(b^*_{\alpha,\beta}) \cap C$ to X. This adds at most d vertices of $\mathcal{T}$ to X, while adds to X all vertices of $N(b) \cap C$ such that $b \in B_{\alpha,\beta}$ and $b^*_{\alpha,\beta} \leq_2 b$. If $\beta > 0$, guess two elements $c^1_{\alpha,\beta}, c^2_{\alpha,\beta} \in N(b^*_{\alpha,\beta}) \cap C \cap \mathcal{T}^\beta$ and add $N(c^1_{\alpha,\beta}) \cap N(c^2_{\alpha,\beta}) \cap B$ to X. This adds at most $(\beta - 1)$ vertices of $\mathcal{T}$ to X, while adds to X every $b \in B_{\alpha,\beta}$ such that $b^*_{\alpha,\beta} \leq_2 b$.

Finally, add the resulting set X to $\mathcal{F}$ if it satisfies the first bullet point of the statement.

We have already argued that in the branch when all guesses concerning $\mathcal{T}$ are correct, the first bullet point of the statement is satisfied. Furthermore, the size of $\mathcal{T} \cap X$ is bounded by

$$\sum_{\alpha=1}^{d} \sum_{\beta=1}^{d} (\alpha + \beta) = 2d\, \frac{d(d+1)}{2} = d^2(d+1).$$

Finally, observe that the number of branches is bounded by $n^{4(d+1)^2}$ as for every $0 \leq \alpha, \beta \leq d$ we guess at most four vertices of G. ☐

Thanks to Lemma 4.8, Lemma 4.9 is applicable in the following setting.

Definition 4.10. Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2 and let $Z \subseteq V(G)$. The *neighborhood partition with respect to* Z is a partition $\mathcal{Z}$ of $V(G) \setminus Z$ into sets depending on (1) their side in the bipartition, and (2) their neighborhood in Z. More precisely,

$$\mathcal{Z} = \{A_{i,Y} \mid i \in \{1,2\}, Y \subseteq Z \cap V_{3-i}\},$$

where

$$A_{i,Y} = \{v \in (V(G) \setminus Z) \cap V_i \mid N(v) \cap Z = Y\}.$$

Lemma 4.11. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2, let $Z \subseteq V(G)$, and let $\mathcal{Z}$ be the neighborhood partition with respect to Z. Then, for every distinct $A, B, C \in \mathcal{Z}$, the triple (A, B, C) is $\mathbb{W}$-free.*

Proof. There is no P_3 of the form $A - B - C$ unless A and C are on one side of the bipartition, and B is on the other side of the bipartition. Then, by the definition of $\mathcal{Z}$, the vertices of A and the vertices of C differ in their neighborhood in Z: there exists $v \in Z$ such that either $N(v) \cap (A \cup C) = A$ or $N(v) \cap (A \cup C) = C$. Then, the claim follows from Lemma 4.8 and the symmetry of A, C. ☐

We summarize this section with the following cleaning step. We remark that if G is connected, the assumption of a fixed bipartition is not needed, as a connected bipartite graph has only one bipartition.

Lemma 4.12. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2, let $Z \subseteq V(G)$, and let $\mathcal{Z}$ be the neighborhood partition with respect to Z. Then one can in time $|V(G)|^{2^{3|Z|}\cdot\mathcal{O}(d^2)}$ enumerate a family $\mathcal{F}$ of at most $n^{4(d+1)^2 2^{3|Z|}}$ subsets of $V(G) \setminus Z$ with the following properties:*

- *For every $X \in \mathcal{F}$, for every $v \in V(G) \setminus X$, the elements of $N(v) \setminus (X \cup Z)$ are contained in a single set of $\mathcal{Z}$.*
- *For every treedepth-d structure $\mathcal{T}$ in G, there exists $X \in \mathcal{F}$ such that $|X \cap \mathcal{T}| \leq 2^{3|Z|} \cdot d^2(d+1)$.*

Proof. Observe that $|\mathcal{Z}| \leq 2^{|Z|} + 1$. Let $\mathcal{Z}^3$ be the family of all triples (A, B, C) where A, B, C are distinct elements of $\mathcal{Z}$. For every triple $(A, B, C) \in \mathcal{Z}^3$, we apply Lemma 4.9 obtaining a family $\mathcal{F}_{(A,B,C)}$; Lemma 4.11 asserts that the assumptions are satisfied. For every triple $(X_{(A,B,C)})_{(A,B,C)\in\mathcal{Z}^3} \in \prod_{(A,B,C)\in\mathcal{Z}^3} \mathcal{F}_{(A,B,C)}$ insert $\bigcup_{(A,B,C)\in\mathcal{Z}^3} X_{(A,B,C)}$ into $\mathcal{F}$. The promised guarantees and size bounds are immediate from Lemma 4.9 and the bound $|\mathcal{Z}| \leq 2^{|Z|+1}$, which implies $|\mathcal{Z}^3| \leq (2^{|Z|} + 1) \cdot 2^{|Z|} \cdot (2^{|Z|} - 1) \leq 2^{3|Z|}$. ☐

4.3 Structural properties of a bad C_6

The tools developed in Section 4.1 allow us to solve $(\mathrm{td} \leq d, \phi)$-MWIS on G, assuming that there is no bad C_6 w.r.t. the solution treedepth-d structure $\mathcal{T}$. In this section we investigate properties of bad C_6s.

We will often denote an induced C_6 as $C = c_1 - c_2 - \ldots - c_6 - c_1$. We implicitly assume that the indices behave cyclically, i.e., $c_7 = c_1$ etc. Then (c_1, c_4), (c_2, c_5), (c_3, c_6) are pairs of the opposite vertices in C.

We will need the following observation implicit in [6]; here we provide a proof for completeness.

Lemma 4.13. *Let G be a P_7-free bipartite graph, let $C = c_1 - c_2 - \ldots - c_6 - c_1$ be an induced C_6 in G, and let D be a connected component of $G - N[V(C)]$ that contains at least two vertices. Then, for every $v \in N(D)$, the set $N(v) \cap V(C)$ equals to either $\{c_1, c_3, c_5\}$ or $\{c_2, c_4, c_6\}$.*

Proof. Let $v \in N(D)$. Since $|D| > 1$, there exists a P_3 of the form $v - D - D$. By the definition of D, $v \in N(V(C))$. Observe that if $c_i \in N(v)$, then $c_{i+2} \in N(v)$, as otherwise G contains the following P_7: $c_{i+3} - c_{i+2} - c_{i+1} - c_i - v - D - D$. The lemma follows. $\square$

For an induced six-vertex-cycle $C = c_1 - c_2 - \ldots - c_6 - c_1$, denote

$$S_1^C = \{v \in N(V(C)) \mid N(v) \cap V(C) = \{c_1, c_3, c_5\}\},$$
$$S_2^C = \{v \in N(V(C)) \mid N(v) \cap V(C) = \{c_2, c_4, c_6\}\}.$$

Furthermore, let MR_C be the union of vertex sets of all connected components of $G - N[V(C)]$ that contain at least 2 vertices. (MR stands for the "main remainder".) With this notation, Lemma 4.13 states that $N(\mathsf{MR}_C) \subseteq S_1^C \cup S_2^C$.

Greatly simplifying, our algorithm will guess a bad C_6 (C, x, y), resolve $N[V(C)]$ using cleaning, and recurse on connected components of MR_C. To restrict the space of possible recursive calls, we need the following observation.

Lemma 4.14. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2, and let $\mathcal{B}$ be a family of pairwise disjoint and anticomplete subsets of $V(G)$ such that for every $B \in \mathcal{B}$, $G[B]$ is connected and $|B| > 1$. Let D be a connected component of $G - \bigcup_{B \in \mathcal{B}} N[B]$, let C be the connected component of G that contains D. Then, for every $i \in \{1, 2\}$, either $N(D) \cap V_i = \emptyset$ or there exists a single set $B_i \in \mathcal{B}$ such that $N(D) \cap V_i \subseteq N(B_i)$. Consequently, there exists a subfamily $\mathcal{B}' \subseteq \mathcal{B}$ of size at most 2 such that $N(D) \subseteq \bigcup_{B \in \mathcal{B}'} N[B]$ (i.e., D is a connected component of $G - \bigcup_{B \in \mathcal{B}'} N[B]$).*

Proof. Let $\mathcal{B}_i \subseteq \mathcal{B}$ be an inclusion-wise minimal set such that $\bigcup_{B \in \mathcal{B}_i} N(B) \supseteq N(D) \cap V_i$. Assume $|\mathcal{B}_i| > 1$; let $B, B' \in \mathcal{B}_i$ be distinct. By minimality, there exists $v, v' \in N(D) \cap V_i$ such that $v \in N(B) \setminus N(B')$ and $v' \in N(B') \setminus N(B)$. Since $|B|, |B'| > 1$, there exists an induced P_3 of the form $v - B - B$ and an induced P_3 of the form $v' - B' - B'$. Let Q be a shortest path from v to v' via D. Then, there exists an induced path in G of the form $B - B - v - Q - v' - B' - B'$ and this path has at least 7 vertices, a contradiction. $\square$

Let $\mathcal{T}$ be a treedepth-d structure in G. A set $A \subseteq V(G)$ is *$\mathcal{T}$-rich* if there exists $\alpha \in [d]$ such that $|A \cap \mathcal{T}^\alpha| > 1$, and *$\mathcal{T}$-poor* otherwise. For $\alpha \in [d]$ and $v \in \mathcal{T}^\alpha$, we denote by $\mathsf{depth}(v) = \alpha$ the depth of v. The *depth* of a bad C_6 (C, x, y) is the pair $\mathsf{depth}(C, x, y) := (\mathsf{depth}(x), \mathsf{depth}(y)) \in [d] \times [d]$. As the depth of a bad C_6 is a pair, in order to compare depths of bad C_6s, we introduce a total order $\prec$ on $[d] \times [d]$ as the lexicographical comparison of $(\mathsf{depth}(x) + \mathsf{depth}(y), \mathsf{depth}(x))$, i.e.,

$$(1,1) \prec (1,2) \prec (2,1) \prec (1,3) \prec (2,2) \prec (3,1) \prec \ldots \prec (d-1,d) \prec (d,d-1) \prec (d,d).$$

A bad C_6 is *$\prec$-maximal* if there is no other bad C_6 whose depth is larger in the $\prec$ order.

We have the following observation.

Lemma 4.15. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2, and let $\mathcal{T}$ be a treedepth-d structure in G. Let (C, x, y) be a bad C_6 with $\mathsf{depth}(C, x, y) = (\alpha, \beta)$. If S_1^C contains at least two vertices of $\mathcal{T}^{\alpha'}$ for some $\alpha' \in [d]$, then $\alpha' > \alpha$. Similarly, if S_2^C contains at least two vertices of $\mathcal{T}^{\beta'}$ for some $\beta' \in [d]$, then $\beta' > \beta$.*

Furthermore, if both S_1^C and S_2^C are $\mathcal{T}$-rich, then there exists a bad C_6 in G of depth (β', α') for some $\alpha' > \alpha$ and $\beta' > \beta$. Consequently, if (C, x, y) is a $\prec$-maximal bad C_6, then either S_1^C or S_2^C is $\mathcal{T}$-poor (possibly both).

Proof. Let $C = c_1 - c_2 - \ldots - c_6 - c_1$ where $c_1 = x$ and $c_4 = y$. Assume that S_1^C contains two vertices $x', x'' \in \mathcal{T}^{\alpha'}$; as $xx', xx'' \in E(G)$ and $x \in \mathcal{T}^{\alpha}$, we have $\alpha' > \alpha$ and x', x'' are descendants of x in $\mathcal{T}$. Similarly, if S_2^C contains two vertices $y', y'' \in \mathcal{T}^{\beta'}$, then $\beta' > \beta$ and y', y'' are descendants of y in $\mathcal{T}$. This proves the first part of the lemma.

If both of the above happen, then since x and y are incomparable in $\mathcal{T}$, x' and y' are also incomparable in $\mathcal{T}$. Then, $C' = x' - c_3 - c_2 - y' - c_6 - c_5 - x'$ is an induced C_6 and (C', y', x') is a bad C_6 of depth (β', α'). As $\alpha' > \alpha$ and $\beta' > \beta$, we have that $\alpha' + \beta' > \alpha + \beta$ and thus $(\alpha, \beta) \prec (\beta', \alpha')$. ☐

We will need the following nomenclature. The *class* of a bad C_6 (C, x, y) (with respect to some fixed treedepth-d structure $\mathcal{T}$) is the pair of bits indicating whether S_1^C is $\mathcal{T}$-poor and whether S_2^C is $\mathcal{T}$-poor. Let Cls be the set of possible classes (i.e., $|\mathsf{Cls}| = 4$). We will usually denote them as poor/poor, rich/poor, poor/rich, and rich/rich.

Informally speaking, if (C, x, y) is a $\prec$-maximal bad C_6 that is of poor/poor class with respect to $\mathcal{T}$ (where $(\mathcal{T}, X)$ is the sought solution), then it is relatively easy to separate $N[V(C)]$ from MR_C, as $S_1^C \cup S_2^C$ contains at most $2d$ vertices of the solution. However, if S_1^C or S_2^C is $\mathcal{T}$-rich, the situation requires a bit more work. The next few lemmata investigate such a situation.

Lemma 4.16. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2, let $Z \subseteq V(G)$ be connected, and let D, D' be two connected components of $G - N[Z]$ that are of size at least 2 each. Then, for every $i \in \{1, 2\}$, the sets $N(D) \cap V_i$ and $N(D') \cap V_i$ are comparable by inclusion.*

Proof. Assume the contrary, let $x \in (N(D) \setminus N(D')) \cap V_i$ and $x' \in (N(D') \setminus N(D)) \cap V_i$. By the existence of x and x', we observe that Z, x, x', D, D' lie all in the same connected component of G and $x, x' \in N(Z)$. Let Q be the shortest path from x to x' via Z. Since $|D|, |D'| > 1$, there exists an induced P_3 of the form $x - D - D$ and an induced P_3 of the form $x' - D' - D'$. But then there exists an induced path of the form $D - D - x - Q - x' - D' - D'$, which has at least 7 vertices, a contradiction. ☐

The most frequent usage of Lemma 4.16 will be when $Z = V(C)$ for some bad C_6 (C, x, y). Then, D, D' are connected components of MR_C and Lemma 4.16 asserts that for every $i \in \{1, 2\}$, the neighborhoods $N(D) \cap S_i^C$ and $N(D') \cap S_i^C$ are comparable by inclusion.

Lemma 4.17. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2. Let $\mathcal{T}$ be a treedepth-d structure in G and let (C, x_1, x_2) be a bad C_6 w.r.t. $\mathcal{T}$. Let $(\alpha_1, \alpha_2) = \mathsf{depth}(C, x_1, x_2)$. Let $i \in \{1, 2\}$ and assume that S_{3-i}^C contains a vertex of $\mathcal{T}^{>\alpha_i}$.*

A bad C_6 (C', x_1', x_2') is called similar *to (C, x_1, x_2) if (a) C is disjoint and anticomplete to C', (b) $\mathsf{depth}(C', x_1', x_2') = \mathsf{depth}(C, x_1, x_2)$, and (c) the component D' of $G[\mathsf{MR}_C]$ that contains C' is adjacent to at least one vertex of $S_{3-i}^C \cap \mathcal{T}^{>\alpha_i}$,*

Then, for every bad C_6 (C', x_1', x_2') that is similar to (C, x_1, x_2), the following holds.

1. *C' is anticomplete to $S_{3-i}^C \cap \mathcal{T}^{>\alpha_i}$.*
2. *For every $u \in S_{3-i}^C \cap \mathcal{T}^{>\alpha_i} \cap N(D')$ there exists $v \in D'$ that is a neighbor of u and belongs to $S_i^{C'}$.*
3. *For every other bad C_6 (C'', x_1'', x_2'') that is similar to (C, x_1, x_2), C'' is contained in D'.*

Proof. As C and C' are disjoint and anticomplete, C' lies in MR_C and C lies in $\mathsf{MR}_{C'}$.

Consequently, if $u \in N(D')$ has a neighbor in C', then $u \in S_1^{C'} \cup S_2^{C'}$, that is, u is adjacent to all three vertices of C' from the same bipartition side. If additionally $u \in S_{3-i}^C$, then u is adjacent to both x_i and x'_i; as these vertices are both in $\mathcal{T}^{\alpha_i}$, u cannot belong to $\mathcal{T}^{>\alpha_i}$. This proves the first point.

Let $u \in N(D') \cap \mathcal{T}^{>\alpha_i} \cap S_{3-i}^C$ and let Q be a shortest path from u to $V(C')$ via D'. Let v be the penultimate (i.e., just before the endpoint in $V(C')$) vertex of Q. The first point asserts that u is anticomplete to C', i.e., $v \neq u$. Since $u \in N(V(C))$ and $C \subseteq \mathsf{MR}_{C'}$, by Lemma 4.13, $v \in S_1^{C'} \cup S_2^{C'}$. Note that there exists an induced P_3 of the form $v - C' - C'$ and an induced P_3 of the form $u - C - C$. Then, G contains an induced path $C' - C' - v - Q - u - C - C$. Since G is P_7-free, Q is of length one, i.e., $vu \in E(G)$. As u and v are on different sides of the bipartition, this proves the second point.

For the third point, let (C'', x''_1, x''_2) be a bad C_6 w.r.t. $\mathcal{T}$ that is similar to (C, x_1, x_2), but contained in a component $D'' \neq D'$ of $G[\mathsf{MR}_C]$. Pick any $u' \in S_{3-i}^C \cap \mathcal{T}^{>\alpha_i} \cap N(D')$ and let $v' \in D'$ be the vertex whose existence is asserted in the second point for (C', x'_1, x'_2). Pick any $u'' \in S_{3-i}^C \cap \mathcal{T}^{>\alpha_i} \cap N(D'')$ (possibly $u' = u''$) and let $v'' \in D''$ be the vertex whose existence is asserted in the second point for (C'', x''_1, x''_2). Note that there exists an induced P_4 of the form $u' - v' - C' - C'$ and an induced P_4 of the form $u'' - v'' - C'' - C''$. By possibly connecting these two P_4s via C (if $u' \neq u''$), we obtain an induced path in G on at least 7 vertices, a contradiction. This proves the third point. ☐

Lemma 4.18. *Let G be a P_7-free bipartite graph with a fixed bipartition V_1, V_2. Let $\mathcal{T}$ be a treedepth-d structure in G, and let (C, x_1, x_2) and (C', x'_1, x'_2) be two disjoint and anticomplete bad C_6s w.r.t. $\mathcal{T}$ that are in the same connected component of G such that $(\alpha_1, \alpha_2) = \mathsf{depth}(C, x_1, x_2) = \mathsf{depth}(C', x'_1, x'_2)$. Furthermore, let $u \in (S_1^C \cup S_2^C) \setminus N[V(C')]$ and $v \in (S_1^{C'} \cup S_2^{C'}) \setminus N[V(C)]$ with $uv \in E(G)$. Denote $Z = V(C) \cup V(C') \cup \{u, v\}$. Then, for every connected component D of $G - N[Z]$, the following holds.*

1. *If $|D| > 1$, then $N(D) \subseteq N(V(C) \cup V(C'))$.*
2. *if additionally D contains a bad C_6 (C'', x''_1, x''_2) with $\mathsf{depth}(C'', x''_1, x''_2) = (\alpha_1, \alpha_2)$, then $N(D) \cap V_1$ and $N(D) \cap V_2$ are $\mathcal{T}$-poor.*

Proof. Let D be a connected component of $G - N[Z]$ with $|D| > 1$.

Let $w \in N(D)$. By the definition of D, w is adjacent to at least one vertex of Z. If w is anticomplete to both $V(C)$ and $V(C')$, then there exists an induced P_5 of the form $w - v - u - C - C$ or $w - u - v - C' - C'$. If $|D| > 1$, then there exists an induced P_3 of the form $w - D - D$, resulting in a P_7 with the aforementioned P_5, a contradiction. Thus, $N(D) \subseteq N(V(C)) \cup N(V(C'))$, as promised in the first point.

Let now (C'', x''_1, x''_2) be as in the second point and assume $w \in N(D) \cap \mathcal{T}$. Let $i \in \{1, 2\}$ be such that $w \in V_{3-i}$. By Lemma 4.13, w is either anticomplete to C or $w \in S_i^C$ and, similarly, w is either anticomplete to C' or $w \in S_i^{C'}$. We have already established that $w \in N(V(C)) \cup N(V(C'))$, that is, $w \in S_i^C \cup S_i^{C'}$; in particular, w is adjacent to either x_i or x'_i.

Assume first that w is adjacent to at most one vertex of the set $\{x_i, x'_i, x''_i\}$. Then, w is adjacent to at least one vertex of Z, but is not complete to all vertices of Z on the opposite side of the bipartition. Consequently, there exists an induced P_4 of the form $w - Z - Z - Z$. Furthermore, as w is adjacent to x_i or x'_i, we know that w is nonadjacent to x''_i. By Lemma 4.13, w is anticomplete to C''. However, then a shortest path from w to C'' via D, with one more step in C'', gives an induced P_4 of the form $w - D - D - D$. Concatenated with the already established P_4 of the form $w - Z - Z - Z$ this gives a P_7 in G, a contradiction.

Hence, w is adjacent to at least two of the vertices of $\{x_i, x'_i, x''_i\}$. Since all these vertices are of depth α_i in $\mathcal{T}$, w is a common ancestor of all vertices of $N(w) \cap \{x_i, x'_i, x''_i\}$. As the choice of $w \in \mathcal{T} \cap N(D)$ was arbitrary, this proves that $N(D) \cap V_1$ and $N(D) \cap V_2$ are $\mathcal{T}$-poor, as desired. ☐

4.4 The algorithm

Armed with the structural insights from the previous sections, we are ready to present the algorithm of Theorem 4.1. It will be a recursive procedure that takes as input an induced subgraph $G[A]$ of G and attempts at decomposing $G[A]$. (The recursive call will take also a number of additional parameters on input to guide the recursion; these will be described shortly.) Similarly as it was done in [13], we will describe the algorithm as guessing properties of a fixed hypothetical treedepth-d structure $\mathcal{T}$ in G that we want to handle.

We define

$$h_{\max} := 48(d+1)^3,$$
$$d_{\max} := 2^{43}(d+1)^3.$$

Recursion arguments and guarantee. A recursive call, apart from the induced subgraph $G[A]$, takes as input two integers $\alpha_1, \alpha_2 \in [d]$, a subset $\mathsf{ActiveCls} \subseteq \mathsf{Cls}$, and a function $\gamma : \mathsf{ActiveCls} \to \mathbb{N}$. Let h be the recursion depth of the recursive call in question.

The goal is to compute a family $\mathcal{C}$ of subsets of A with the following guarantee for every treedepth-d structure $\mathcal{T}$.

Recursion guarantee. Suppose the following hold:

- There is no bad C_6 (C, x, y) w.r.t. $\mathcal{T}$ with $V(C) \subseteq A$, and $(\alpha_1, \alpha_2) \prec (\mathsf{depth}(x), \mathsf{depth}(y))$.
- For every bad C_6 (C, x, y) w.r.t. $\mathcal{T}$ with $V(C) \subseteq A$, and $(\alpha_1, \alpha_2) = (\mathsf{depth}(x), \mathsf{depth}(y))$, the class c of (C, x, y) belongs to $\mathsf{ActiveCls}$. Furthermore, the number of vertices of $\mathcal{T} \setminus A$ that are adjacent to $V(C)$ is at least $\gamma(\mathsf{c})$.
- $N(A) \cap \mathcal{T}$ is of size at most $d_{\max}^{4^h}$.

Then there exists a tree decomposition (T, β) of $G[A]$ such that for every $t \in V(T)$ we have $\beta(t) \in \mathcal{C}$ and $|\beta(t) \cap \mathcal{T}| \leq d_{\max}^{4^{h+1}}$.

We will maintain the following progress in the recursion. For a call on $G[A]$, (α_1, α_2), $\mathsf{ActiveCls}$, and γ, a direct subcall on $G[A']$, (α'_1, α'_2), $\mathsf{ActiveCls}'$, γ' satisfies:

- $(\alpha'_1, \alpha'_2) \prec (\alpha_1, \alpha_2)$, or
- $(\alpha'_1, \alpha'_2) = (\alpha_1, \alpha_2)$ and
  - $\mathsf{ActiveCls}' \subsetneq \mathsf{ActiveCls}$, or
  - $\mathsf{ActiveCls}' = \mathsf{ActiveCls}$, $\gamma(\mathsf{c}) \leq 12d$ for every $\mathsf{c} \in \mathsf{ActiveCls}$, $\gamma'(\mathsf{c}) \geq \gamma(\mathsf{c})$ for every $\mathsf{c} \in \mathsf{ActiveCls}$, and $\gamma'(\mathsf{c}) > \gamma(\mathsf{c})$ for at least one $\mathsf{c} \in \mathsf{ActiveCls}$.

This progress property will always be immediate from the description of the algorithm and thus usually not checked explicitly. Note that this progress property bounds the depth of the recursion by $(d+1)^2 \cdot (48d+4) \leq h_{\max}$.

Leaves of the recursion. The recursion reaches its leaf when either $G[A]$ is C_6-free (hence chordal bipartite) or we have $(\alpha_1, \alpha_2) = (1, 1)$ and $\mathsf{ActiveCls} = \emptyset$. In both cases, for every $\mathcal{T}$ that satisfies the premise of the recursion guarantee, there is no bad C_6 w.r.t. $\mathcal{T}$. Hence, it suffices to invoke the algorithm of Lemma 4.7 to $G[A]$ and return the computed family $\mathcal{C}$.

Recursion. Let us focus on a recursive call with input $G[A]$, (α_1, α_2), $\mathsf{ActiveCls}$, and γ. If $G[A]$ is disconnected, we recurse on each connected component independently and return the union of the returned families. Hence, we assume that $G[A]$ is connected.

Let $\mathcal{T}$ be a treedepth-d structure in G that satisfies the premise of the recursion guarantee. Thanks to Lemma 4.15, we can delete rich/rich from $\mathsf{ActiveCls}$ and the domain of γ, if it is present there. If $\mathsf{ActiveCls} = \emptyset$, then there are no bad C_6s (C, x, y) of level (α_1, α_2), so we can recurse on the immediate predecessor of (α_1, α_2) in the $\prec$ order, with $\mathsf{ActiveCls} = \mathsf{Cls}$ and $\gamma \equiv 0$, and pass on the set $\mathcal{C}$ returned by this call.

If there exists $\mathsf{c} \in \mathsf{ActiveCls}$ with $\gamma(\mathsf{c}) > 12d$, proceed as follows. Observe that if (C, x, y) is a bad C_6 of level (α_1, α_2) and of class c with $V(C) \subseteq A$, then there is $c \in V(C)$ that is adjacent to at least two vertices in $N(A) \cap \mathcal{T}$. Iterate over all possible choices of $N(A) \cap \mathcal{T}$ (there is only a polynomial number of them thanks to the recursion guarantee premise) together with the levels of vertices of $N(A) \cap \mathcal{T}$. Let B be the union, over all pairs (u, u') of distinct vertices of $N(A) \cap \mathcal{T}$ of the same level, of the sets $N(u) \cap N(u') \cap A$. Recurse on $G[A \setminus B]$, (α_1, α_2), and $\mathsf{ActiveCls} \setminus \{\mathsf{c}\}$, obtaining a set $\mathcal{C}_B$. Finally, return the union of all sets $\{B \cup D \mid D \in \mathcal{C}_B\}$ constructed in this process.

Note that there is only a $(dn)^{d_{\max}^{4^h}} = n^{d_{\max}^{2^{\mathcal{O}(d^3)}}}$ choices of $N(A) \cap \mathcal{T}$ together with its levels and, for every pair (u, u'), the set $N(u) \cap N(u')$ contains at most $(d-1)$ vertices of $\mathcal{T}$. Hence, the set B constructed in the branch where all information on $N(A) \cap \mathcal{T}$ is guessed correctly satisfies $|B \cap \mathcal{T}| \leq (d-1)\binom{d_{\max}^{4^h}}{2} \leq d_{\max}^{4^{h+1}}$. Consequently, in this branch the premise of the recursion guarantee for $\mathcal{T}$ is satisfied for the recursive subcalls; if (T', β') is the tree decomposition promised for $G[A \setminus B]$ and $\mathcal{T}$, then we can construct the promised tree decomposition (T, β) by taking $T = T'$ and $\beta(t) = \beta'(t) \cup B$ for every $t \in V(T')$.

We are left with the case $\gamma(\mathsf{c}) \leq 12d$ for every $\mathsf{c} \in \mathsf{ActiveCls}$. Pick $\mathsf{c} \in \mathsf{ActiveCls}$, preferably rich/poor or poor/rich if one of those belong to $\mathsf{ActiveCls}$. The algorithm will perform an internal dynamic programming routine on a carefully chosen family of subsets of A. We say that a set $A' \subseteq A$ is *well-shaped* if there is a family $\mathcal{B}$ consisting of at most two connected subsets of A, each on at least 2 and at most 14 vertices, such that A' is a connected component of $G[A] - N[\bigcup_{B \in \mathcal{B}} B]$. Let $\mathcal{A}$ be the set of well-shaped sets; note that $|\mathcal{A}| \leq |A|^{29}$ and $\mathcal{A}$ can be computed in polynomial time.

A well-shaped set A' is *$\mathcal{T}$-regular* if there exists a family $\mathcal{B}$ witnessing that A' is well-shaped such that for every $B \in \mathcal{B}$ and $i \in \{1, 2\}$, the set $N(A') \cap A \cap N(B) \cap V_i$ is $\mathcal{T}$-poor. Note that if A' is $\mathcal{T}$-regular, then $|N(A') \cap A \cap \mathcal{T}| \leq 4d$.

Note that A is $\mathcal{T}$-regular (in particular, $A \in \mathcal{A}$), as witnessed by $\mathcal{B} = \emptyset$ (recall that $G[A]$ is connected).

For every $A' \in \mathcal{A}$, in the order of increasing size of A', we will compute a family $\mathcal{C}(A')$ of subsets of $A \cap N[A']$ with the following guarantee: if A' is $\mathcal{T}$-regular, then there exists a tree decomposition (T, β) of $G[N[A'] \cap A]$ such that for every $t \in V(T)$ we have $\beta(t) \in \mathcal{C}(A')$ and $|\beta(t) \cap \mathcal{T}| \leq d_{\max}^{4^{h+1}}$. The family $\mathcal{C}(A)$ is the desired output of the recursive call.

Lemma 4.14 ensures that the following process does not lead outside $\mathcal{A}$.

Claim 4.19. *Let $A' \in \mathcal{A}$, let B be a connected subset of $G[A']$ on at least 2 and at most 14 vertices, and let D be a connected component of $G[A' \setminus N[B]]$. Then, $D \in \mathcal{A}$. Furthermore, if A' is $\mathcal{T}$-regular and, for every $\alpha \in \{1, 2\}$, the set $N(D) \cap A \cap N(B) \cap V_\alpha$ is $\mathcal{T}$-poor, then D is $\mathcal{T}$-regular, too.*

Proof. Let $\mathcal{B}$ be a family witnessing $A' \in \mathcal{A}$. Let $\mathcal{B}' = \mathcal{B} \cup \{B\}$. Then, D is a connected component of $G[A] - \bigcup_{B' \in \mathcal{B}'} N[B']$ and the family $\mathcal{B}'$ satisfies the requirements of Lemma 4.14 in the graph $G[A]$. Hence, there is a subfamily $\mathcal{B}'' \subseteq \mathcal{B}'$ of size at most 2 such that D is a connected component of $G[A] - \bigcup_{B'' \in \mathcal{B}''} N[B'']$. Then, $\mathcal{B}''$ witnesses that $D \in \mathcal{A}$.

For the second statement, observe that if $\mathcal{B}$ witnesses that A' is $\mathcal{T}$-regular, then so does $\mathcal{B}''$ for D. ☐

We emphasize here that the set $N(D) \cap A \cap N(B) \cap V_i$ in Claim 4.19 may contain vertices outside A' and our control of the number of vertices of $\mathcal{T}$ in this set is very important for the correctness of the algorithm.

Fix $A' \in \mathcal{A}$. A bad C_6 (C, x, y) of level (α_1, α_2) and of class c with $V(C) \subseteq A'$ is henceforth called a *relevant* C_6.

Guess if $\mathcal{T}$ contains a relevant C_6. For the "no" branch, recurse on $G[A']$, (α_1, α_2), and $\mathsf{ActiveCls} \setminus \{\mathsf{c}\}$, and put every returned set into the computed family $\mathcal{C}(A')$.

For the “yes” branch, proceed as follows. For a connected set $A'' \subseteq A'$, a relevant C_6 (C, x, y) in $G[A'']$ is *extremal* if it maximizes the number of vertices in its main remainder in $G[A'']$ and, subject to the above, it minimizes the number of vertices of $\mathcal{T}$ in $N[V(C)] \cap A''$.

Finding a pivot set Z. We will now identify (by branching) a connected set $Z \subseteq A'$ with the following properties.

- Z is of one of the following forms:

  (single C_6) $V(C)$, where (C, x, y) is an extremal relevant C_6 in $G[A']$, or

  (double C_6) $V(C) \cup V(C') \cup \{u, v\}$, where (C, x, y) is an extremal relevant C_6 in $G[A']$, (C', x', y') is a relevant C_6 that is extremal in a connected component D' of $G[A'] - N[V(C)]$, $u \in S^C_i \cap A' \cap \mathcal{T}$ for some $i \in \{1, 2\}$, $u \notin N(V(C'))$, $v \in S^{C'}_{3-i} \cap D'$, and $uv \in E(G)$.

- For every connected component F of $G[A'] - N[Z]$ that contains a relevant C_6, for every $i \in \{1, 2\}$, the set $N(F) \cap A \cap N(Z) \cap V_\alpha$ is $\mathcal{T}$-poor.

We start by guessing an extremal relevant C_6 (C, x, y). Somewhat abusing the notation, let us denote its main remainder in $G[A']$ by MR_C.

We check if $Z = V(C)$ satisfies the above properties. If this is the case, we are done, so assume otherwise.

We say that a component D of $G[\mathsf{MR}_C]$ is *terrifying* if it contains a relevant C_6 and additionally $N(D) \cap A \cap N(Z) \cap V_1$ is $\mathcal{T}$-rich or $N(D) \cap A \cap N(Z) \cap V_2$ is $\mathcal{T}$-rich. Observe that $Z = V(C)$ does not satisfy the desired properties if and only there exists a terrifying component. Note that it can happen only if c is not poor/poor.

Let D_0 be a terrifying component. Guess D_0 and a relevant C_6 (C', x', y') that is extremal in $G[D_0]$. Lemma 4.15 implies that $N(D_0) \cap A \cap S^C_1$ contains two vertices of $\mathcal{T}^{\alpha'_1}$ for some $\alpha'_1 > \alpha_1$ or $N(D_0) \cap A \cap S^C_2$ contains two vertices of $\mathcal{T}^{\alpha'_2}$ for some $\alpha'_2 > \alpha_2$; let u be any of those two vertices. Then, Lemma 4.17 applies to the graph $G[A' \cup (A \cap (S^C_1 \cup S^C_2))]$: by the first point, $u \notin N(V(C'))$, the second point gives $v \in D_0 \cap (S^{C'}_1 \cup S^{C'}_2)$ with $uv \in E(G)$, and the third point implies that D_0 is the only terrifying component.

Denote $Z := V(C) \cup V(C') \cup \{u, v\}$. Lemma 4.18 applies to (C, x, y), (C', x', y'), u, and v in the graph $G[A' \cup (A \cap N[Z])]$: every connected component D of $G[A'] - N[Z]$ that contains a relevant C_6, the set $N(D) \cap A \cap N(Z)$ is contained in $N(V(C)) \cup N(V(C'))$ and both $N(D) \cap A \cap N(Z) \cap V_1$ and $N(D) \cap A \cap N(Z) \cap V_2$ are $\mathcal{T}$-poor. Consequently, Z satisfies the desired properties.

Finding a neighborhood Y. Lemma 4.16, applied to Z in the graph $G[A' \cup (A \cap N[Z])]$ asserts that the components of $G[A'] - N[Z]$ that contain a relevant C_6 have comparable neighborhoods in $A \cap N[Z] \cap V_1$ and comparable neighborhoods in $A \cap N[Z] \cap V_2$. For every $i \in \{1, 2\}$, guess a component D°_i of $G[A'] - N[Z]$ that contains a relevant C_6 and with maximal $N(D^\circ_i) \cap A \cap N[Z] \cap V_i$; a valid guess is $D^\circ_i = \emptyset$ if no such component exists. By the properties of Z, we have that $N(D^\circ_i) \cap A \cap N[Z] \cap V_i$ is $\mathcal{T}$-poor.

Let $Y := \bigcup_{i \in \{1,2\}} N(D^\circ_i) \cap A \cap N[Z] \cap V_i$. We have that $|Y \cap \mathcal{T}| \leq 2d$ and for every component D of $G[A'] - N[Z]$, if D contains a relevant C_6, then $N(D) \cap A \cap N[Z] \subseteq Y$.

Let D_{poor} be the union of vertex sets of all components D of $G[A'] - N[Z]$ such that $|D| > 1$ and $N(D) \cap A \cap N[Z] \subseteq Y$. Note that every relevant C_6 that is contained in $G[A'] - N[Z]$ is in fact in $G[D_{\text{poor}}]$.

Using cleaning. Let $\mathcal{F}$ be the result of applying Lemma 4.12 to $G[A' \setminus (Y \cup D_{\text{poor}})]$ and Z. The following claim establishes the progress of the recursion.

Claim 4.20. *For every $X \in \mathcal{F}$, every connected component D of $G[A' \setminus (X \cup Y \cup Z \cup D_{\text{poor}})]$ either does not contain a relevant C_6, or every relevant C_6 in D has at least one neighbor in $(X \cup Y \cup Z) \cap \mathcal{T}$.*

Proof. Let $\mathcal{Z} = \{A_{i,Q} \mid i \in \{1,2\}, Q \subseteq Z \cap V_i\}$ be the neighborhood partition with respect to Z in $G[A' \setminus D_{\text{poor}}]$. Fix $X \in \mathcal{F}$ and consider a connected component D of $G[A' \setminus (X \cup Y \cup Z \cup D_{\text{poor}})]$.

If D is anticomplete to Z, then D does not contain a relevant C_6 as D is not part of D_{poor}. Lemma 4.9 ensures that D is contained in the union of two sets of $\mathcal{Z}$, denote them A_{1,Q_1} and A_{2,Q_2}. If any of those sets is adjacent to a vertex of $Z \cap \mathcal{T}$, then every relevant C_6 in D has a neighbor in $Z \cap \mathcal{T}$, as desired. Otherwise, both A_{1,Q_1} and A_{2,Q_2} are anticomplete to x and y, and, in case Z has the double C_6 structure, also to u, x', and y'.

In particular, D is disjoint with $S_1^C \cup S_2^C$. Hence, either $D \subseteq A' \setminus (\mathsf{MR}_C \cup S_1^C \cup S_2^C)$ or $D \subseteq \mathsf{MR}_C$. Consider now the first case.

If D does not contain a relevant C_6, then we are done, so assume otherwise. Let (C'', x'', y'') be a relevant C_6 in D. Observe that MR_C is contained in the main remainder of C'' in $G[A']$. Hence, as (C, x, y) is extremal, these two main remainders are equal.

Denote $C = c_1 - c_2 - \ldots - c_6 - c_1$ with $c_1 = x$ and $c_4 = y$. Observe that, unless $Q_1 = \{c_3, c_5\}$ and $Q_2 = \{c_2, c_6\}$, there exists an an edge of C that is anticomplete to C''. This edge is contained in the main remainder of C'' in $G[A']$, a contradiction to the fact that (C, x, y) is extremal. In the other case, the fact that (C, x, y) is extremal and (C, x, y) and (C'', x'', y'') have equal main remainders in $G[A']$ implies that $|A' \cap N[V(C'')] \cap \mathcal{T}| \geq |A' \cap N[V(C)] \cap \mathcal{T}|$. Observe that $x, y \in N[V(C)] \setminus N[V(C'')]$. Hence, there are at least two vertices of $\mathcal{T}$ in $A' \cap \mathcal{T} \cap (N[V(C')] \setminus N[V(C)])$. Since $D \subseteq A_{1,Q_1} \cup A_{2,Q_2} \subseteq N(V(C))$, these two vertices of $\mathcal{T}$ necessarily belong to $X \cup Y \cup Z$, as desired. This completes the case $D \subseteq A' \setminus (\mathsf{MR}_C \cup S_1^C \cup S_2^C)$.

We are left with the remaining case $D \subseteq \mathsf{MR}_C$. If Z is of single C_6 structure, then we are done, as $D \cap D_{\text{poor}} = \emptyset$ and thus D contains no relevant C_6. Assume then Z is of double C_6 structure.

Assume first that $D \cap D_0 = \emptyset$. Recall that $u \in Q_1 \cup Q_2$ as otherwise any relevant C_6 in D has a neighbor in $Z \cap \mathcal{T}$ as desired. Hence, $D \cap N[Z] = \emptyset$. Then, as D is disjoint with D_{poor}, D does not contain any relevant C_6.

In the second case, $D \subseteq D_0$ as D is disjoint with $S_1^C \cup S_2^C$. Since $x', y', u \notin Q_1 \cup Q_2$, $D \subseteq D_0 \cap N(V(C')) \setminus (S_1^{C'} \cup S_2^{C'})$ or D is part of the main remainder of C' in D_0. In the latter case, the properties of Z ensure that D is contained in $G[A'] - N[Z]$ and, as $D \cap D_{\text{poor}} = \emptyset$, D contains no relevant C_6.

In the former case, let $C' = c'_1 - c'_2 - \ldots - c'_6 - c'_1$ and assume there exists a relevant C_6 (C'', x'', y'') in D. Observe that the main remainder of C' in $G[D_0]$ is contained in the main remainder of C''. Since (C', x', y') is extremal in $G[D_0]$, these main remainders are equal. Observe that unless $Q_1 \cap V(C') = \{c'_3, c'_5\}$ and $Q_2 \cap V(C') = \{c'_2, c'_6\}$, there is an edge of C' that is anticomplete to C'', a contradiction to the fact that the main remainders of C' and C'' in $G[D_0]$ are equal. Otherwise, since (C', x', y') is extremal, there are at least two elements of $\mathcal{T}$ in $D_0 \cap (N(V(C'')) \setminus N(V(C')))$. Since $D \subseteq N(V(C'))$ (if $Q_1 \cap V(C') = \{c'_3, c'_5\}$ and $Q_2 \cap V(C') = \{c'_2, c'_6\}$), these two elements of $\mathcal{T}$ necessarily are in $X \cup Y \cup Z$, as desired. This completes the case analysis and the proof of the claim. ☐

Wrapping up. We guess Z and Y as described above, compute D_{poor} and invoke Lemma 4.12 to Z in $G[A' \setminus (Y \cup D_{\text{poor}})]$, obtaining a family $\mathcal{F}$. Claim 4.19 ensures that every connected component of $G[D_{\text{poor}}]$ belongs to $\mathcal{A}$ and, furthermore, if A' is $\mathcal{T}$-regular, then so is every connected component of $G[D_{\text{poor}}]$. If this is the case, let (T_D, β_D) be the promised tree decomposition for a component D of $G[D_{\text{poor}}]$. Insert into $\mathcal{C}(A')$ all sets of $\mathcal{C}(D)$ for every connected component D of $G[D_{\text{poor}}]$.

For every $X \in \mathcal{F}$, we proceed as follows. First, insert $X \cup Y \cup Z \cup (N(A') \cap A)$ into $\mathcal{C}(A')$. Then, recurse on every connected component D of $G[A' \setminus (X \cup Y \cup Z \cup D_{\text{poor}})]$ with the same parameters (α_1, α_2), $\mathsf{ActiveCls}$, and γ except for the value $\gamma(\mathsf{c})$ increased by one, obtaining a family $\mathcal{C}'(D)$. For every set $B \in \mathcal{C}'(D)$ from the returned family, insert $X \cup Y \cup Z \cup (N(A') \cap A) \cup B$ into the set $\mathcal{C}(A')$. This completes the description of the algorithm.

Observe that if A' is $\mathcal{T}$-regular, then for the set $X \in \mathcal{F}$ that satisfies the promise of Lemma 4.12 for $\mathcal{T}$, we have

$$|(X \cup Y \cup Z \cup (N(A') \cap A)) \cap \mathcal{T}| \leq 2^{3 \cdot 14} d^2 (d+1) + 2d + 14 + 4d < 2^{43}(d+1)^3 = d_{\max}. \quad (1)$$

Hence, for that set the premise of the recursion guarantee is satisfied, and we have the promised tree decomposition (T_D, β_D) of $G[D]$.

Construct the desired tree decomposition (T, β) as follows. Start with the root bag $(X \cup Y \cup Z \cup (N(A') \cap A))$. For every connected component D of $G[D_{\text{poor}}]$, attach (T_D, β_D) as a subtree with its root adjacent to the root bag of (T, β). For every connected component D of $G[A' \setminus (X \cup Y \cup Z \cup D_{\text{poor}})]$, denote $\beta'_D(t) = \beta_D(t) \cup X \cup Y \cup Z \cup (N(A') \cap A)$ for $t \in V(T_D)$ and attach (T_D, β'_D) as a subtree with its root adjacent to the root bag of (T, β). It is immediate that (T, β) is a tree decomposition of $G[N[A'] \cap A]$; the bound on $|\mathcal{T} \cap \beta(t)|$ for any $t \in V(T)$ follows from (1).

5 Proof of Theorem 1.1

In this section we use Theorems 3.1 and 4.1 to prove Theorem 1.1. Fix an integer $d > 0$.

On the very high level, the algorithm works as follows. We invoke Theorem 3.1 on the input graph G, obtaining the family $\mathcal{F}$. Let $\mathcal{T}$ be the sought solution and let S be a $\mathcal{T}$-avoiding minimal separator of G. Let $\Psi = (K, \mathcal{D}, L) \in \mathcal{F}$ be the element promised for S by Theorem 3.1. Note that $S \subseteq W(\Psi) := K \cup \bigcup \mathcal{D}$. We would like to first understand the components of $\mathcal{D}$ and then use $W(\Psi)$ as a container for S. Luckily, every $D \in \mathcal{D}$ is somewhat easier than the whole graph G. Let $\mathcal{C}(\Psi)$ be the family of all connected components of $G[L(D)]$ and all connected components of $G[D \setminus L(D)]$; recall that they are modules of $G[D]$. Every connected component of $\mathcal{C}(\Psi)$ has clique number strictly smaller than G (so we can recurse on them), while the quotient graph after contracting every component $\mathcal{C}(\Psi)$ in $G[D]$ is bipartite and thus handled by Theorem 4.1.

Unfortunately, we do not know how to handle the aforementioned recursion into $\mathcal{C}(\Psi)$ using only the language of families of containers or carvers and we will need to do it using the language of "partial solutions to CMSO_2 formulae". To this end, we will use the handy notion of a *threshold automata* as in [13].

Threshold automata. Let Σ be a finite alphabet. A Σ*-labelled forest* is a rooted forest F where every $v \in V(F)$ as a label $\mathsf{label}(v) \in \Sigma$.

We use the notation $\{\{\cdot\}\}$ for defining multisets. For a multiset X and an integer $\tau \in \mathbb{N}$, let $X \wedge \tau$ be the multiset obtained from X by the following operation: for every element e whose multiplicity k is larger than 2τ, we reduce its multiplicity to the unique integer in $\{\tau + 1, \dots, 2\tau\}$ with the same residue as k modulo τ. For a set Q, let $\mathsf{Multi}(Q, \tau)$ be the family of all multisets that contain elements of Q, and each element is contained at most 2τ times.

Definition 5.1 ([13]). A *threshold automaton* is a tuple $\mathcal{A} = (Q, \Sigma, \tau, \delta, C)$, where:

- Q is a finite set of states;
- Σ is a finite alphabet;
- $\tau \in \mathbb{N}$ is a nonnegative integer called the *threshold*;
- $\delta \colon \Sigma \times \mathsf{Multi}(Q, \tau) \to Q$ is the transition function; and
- $C \subseteq \mathsf{Multi}(Q, \tau)$ is the accepting condition.

For a Σ-labelled forest F, the *run* of $\mathcal{A}$ on F is the unique labelling $\xi \colon V(F) \to Q$ satisfying the following property for each $x \in V(F)$:

$$\xi(x) = \delta(\mathsf{label}(x), \{\{\xi(y) \colon y \text{ is a child of } x\}\} \wedge \tau).$$

We say that $\mathcal{A}$ *accepts* F if

$$\{\{\xi(z) \colon z \text{ is a root of } F\}\} \wedge \tau \in C,$$

where ξ is the run of $\mathcal{A}$ on F.

Recall that for a CMSO_2 formula with one vertex set variable ϕ, the $(\phi, \mathrm{td} \leq d)$-MWIS problem, given a vertex-weighted graph G, asks for a pair (Sol, X) with $X \subseteq \mathsf{Sol} \subseteq V(G)$ such that $G[\mathsf{Sol}]$ is of treedepth at most d, $\phi(X)$ is satisfied in $G[\mathsf{Sol}]$, and the weight of X is maximized.

Let (Sol, X) be such that $X \subseteq \mathsf{Sol} \subseteq V(G)$ and $\mathrm{td}(G[\mathsf{Sol}]) \leq d$ and let $\mathcal{T}$ be a treedepth-d structure containing Sol. Let label be a labeling of $V(\mathcal{T})$ that labels $v \in V(\mathcal{T})$ with the following information: (1) its depth in $\mathcal{T}$, (2) which of its ancestors in $\mathcal{T}$ it is adjacent to, and (3) whether $v \in X$ and/or $v \in \mathsf{Sol}$. As discussed in [13], the check whether $\phi(X)$ is satisfied in $G[\mathsf{Sol}]$ can be done with a threshold automaton run on $(\mathcal{T}, \mathsf{label})$. More precisely, there exists a threshold automaton $\mathcal{A}$ (depending only on d and ϕ) such that $\phi(X)$ is satisfied in $G[\mathsf{Sol}]$ if and only if $\mathcal{A}$ accepts $\mathcal{T}$ with labeling label. Thus, instead of focusing on a CMSO_2 formula ϕ, we can fix a threshold automaton $\mathcal{A}$ and look for a treedepth-d structure $\mathcal{T}$ and $X \subseteq \mathsf{Sol} \subseteq \mathcal{T}$ such that $\mathcal{A}$ accepts $\mathcal{T}$ with the aforementioned labeling and, subject to the above, the weight of X is maximized. In what follows, the labeling label will be implicit and not mentioned.

Partial solutions. A treedepth-d structure $\mathcal{T}$ is *neat* if for every non-root node v with a parent u, u is adjacent to at least one descendant of v (possibly v itself). A simple (cf. [13]) argument shows that every treedepth-d structure can be turned into a neat one without increasing the depth of any vertex.

Let $\preceq_1$ be the following quasi-order on sets of vertices of a graph G: $X \preceq_1 Y$ if $|X| > |Y|$ or $|X| = |Y|$ and X precedes Y lexicographically. In [13], a quasi-order $\preceq$ on treedepth-d structures in G was introduced: to compare two treedepth-d structures $\mathcal{T}_1$ and $\mathcal{T}_2$, the following sequence of sets is compared with $\preceq_1$ and the first differing one determines whether $\mathcal{T}_1 \preceq \mathcal{T}_2$: first the set of all vertices, then the set of all depth-1 vertices (i.e., roots), then the set of all depth-2 vertices, $\ldots$, the set of all depth-d vertices. Although two distinct treedepth-d structures $\mathcal{T}_1, \mathcal{T}_2$ can satisfy both $\mathcal{T}_1 \preceq \mathcal{T}_2$ and $\mathcal{T}_2 \preceq \mathcal{T}_1$, the quasi-order $\preceq$ is a total order on the set of neat treedepth-d structures [13].

A *partial solution* in a graph G is a tuple $(\mathcal{T}, X, \mathsf{Sol})$ where $\mathcal{T}$ is a treedepth-d structure in G and $X \subseteq \mathsf{Sol} \subseteq V(\mathcal{T})$. As in [13], we extend $\preceq$ to comparing partial solutions: $(\mathcal{T}, X, \mathsf{Sol}) \preceq (\mathcal{T}', X', \mathsf{Sol}')$ if:

1. the weight of X is larger than the weight of X', or
2. the weights of X and X' are equal, but $X \preceq_1 X'$;
3. $X = X'$, but $\mathsf{Sol} \preceq_1 \mathsf{Sol}'$;
4. $X = X'$ and $\mathsf{Sol} = \mathsf{Sol}'$, but $\mathcal{T} \preceq \mathcal{T}'$.

We will be looking for a $\preceq$-minimal solution $(\mathcal{T}, X, \mathsf{Sol})$ to our $(\phi, \mathrm{td} \leq d)$-MWIS problem on G. As shown in [13], $\mathcal{T}$ is then maximal and neat.

A partial solution $(\mathcal{T}', X', \mathsf{Sol}')$ is an *extension* of a partial solution $(\mathcal{T}, X, \mathsf{Sol})$ if $\mathcal{T}$ is an induced subgraph of $\mathcal{T}'$, every root of $\mathcal{T}$ is a root of $\mathcal{T}'$, and $X' \cap \mathcal{T} = X$ and $\mathsf{Sol}' \cap \mathcal{T} = \mathsf{Sol}$. An extension $(\mathcal{T}', X', \mathsf{Sol}')$ of $(\mathcal{T}, X, \mathsf{Sol})$ is *neat* if for every non-root node $v \in V(\mathcal{T}')$ if $v \notin V(\mathcal{T})$, then there is an edge between the parent of v and some descendant of v in $\mathcal{T}'$.

For a fixed threshold automaton $\mathcal{A} = (Q, \Sigma, \tau, \delta, C)$, a *multistate assignment* of a partial solution $(\mathcal{T}, X, \mathsf{Sol})$ is a function $\xi : \{\emptyset\} \cup V(\mathcal{T}) \to \mathsf{Multi}(Q, \tau)$. We say that an extension $(\mathcal{T}', X', \mathsf{Sol}')$ of $(\mathcal{T}, X, \mathsf{Sol})$ *evaluates* to ξ if for the run ξ' of $\mathcal{A}$ on $(\mathcal{T}', X', \mathsf{Sol}')$ (with the aforementioned natural labeling of $\mathcal{T}'$) it holds that

$$\{\{\xi'(z) \mid z \text{ is a root of } \mathcal{T}' \text{ but not of } \mathcal{T}\}\} \wedge \tau_{\mathcal{A}} = \xi(\emptyset),$$

and, for every $v \in V(\mathcal{T})$,

$$\{\{\xi'(z) \mid z \text{ is a child of } v \text{ in } \mathcal{T}' \text{ but not in } \mathcal{T}\}\} \wedge \tau_{\mathcal{A}} = \xi(v).$$

Intuitively, ξ contains all the information on how $\mathcal{T}' \setminus \mathcal{T}$ contributes to the run of $\mathcal{A}$ on $(\mathcal{T}', X', \mathsf{Sol}')$.

We will be computing set families $\mathcal{F}$ as follows.

Definition 5.2. Let $d, \widehat{d}, \widetilde{d} > 0$ be integers and $\mathcal{A}$ be a threshold automaton. For a graph G and a set $D \subseteq V(G)$, we say that a family $\mathcal{F}$ of subsets of D is a *defect-$\widetilde{d}$ extension-$\widehat{d}$ treedepth-d container family* if for every partial solution $(\mathcal{T}, X, \mathsf{Sol})$ in G with $|V(\mathcal{T})| \leq \widehat{d}$ and for every multistate assignment ξ of $(\mathcal{T}, X, \mathsf{Sol})$, if $(\mathcal{T}', X', \mathsf{Sol}')$ is the $\preceq$-minimal extension of $(\mathcal{T}, X, \mathsf{Sol})$ among extensions that satisfy (1) $V(\mathcal{T}') \setminus V(\mathcal{T}) \subseteq D$ and (2) $(\mathcal{T}', X', \mathsf{Sol}')$ evaluates to ξ, then if $(\mathcal{T}', X', \mathsf{Sol}')$ is actually a neat extension of $(\mathcal{T}, X, \mathsf{Sol})$, then there exists a tree decomposition (T, β) of $G[D]$ such that for every $t \in V(T)$ there exists $A \in \mathcal{F}$ with $\beta(t) \subseteq A$ and $|A \cap \mathcal{T}'| \leq \widetilde{d}$.

Furthermore, the family $\mathcal{F}$ is *exact* if additionally $\beta(t) = A$ in the above.

The dynamic programming algorithm of [13, Section 3] can be expressed as the following algorithmic lemma that we will use as a subroutine.

Lemma 5.3 ([13]). *Let $d, \widehat{d}, \widetilde{d} > 0$ be fixed integers and $\mathcal{A}$ be a fixed threshold automaton. There exists a polynomial-time algorithm that takes on input a graph G, a set $D \subseteq V(G)$, and a family $\mathcal{F}$ that is a defect-$\widetilde{d}$ extension-$\widehat{d}$ treedepth-d container family and outputs, for every partial solution $(\mathcal{T}, X, \mathsf{Sol})$ in G with $|V(\mathcal{T})| \leq \widehat{d}$ and for every multistate assignment ξ of $(\mathcal{T}, X, \mathsf{Sol})$, the $\preceq$-minimal extension $(\mathcal{T}', X', \mathsf{Sol}')$ of $(\mathcal{T}, X, \mathsf{Sol})$ among extensions that evaluate to ξ and $V(\mathcal{T}') \setminus V(\mathcal{T}) \subseteq D$, provided that this is a neat extension.*

The family of Theorem 4.1 is stronger, and we capture it with the following definition.

Definition 5.4. Let $d, \widetilde{d} > 0$ be integers. For a graph G, we say that a family $\mathcal{F}$ of subsets of $V(G)$ is a *defect-$\widetilde{d}$ treedepth-d container family* if for every treedepth-d structure $\mathcal{T}$ in G there exists a tree decomposition (T, β) of G such that for every $t \in V(T)$ there exists $A \in \mathcal{F}$ with $\beta(t) \subseteq A$ and $|A \cap \mathcal{T}'| \leq \widetilde{d}$.

Furthermore, the family $\mathcal{F}$ is *exact* if additionally $\beta(t) = A$ in the above.

In both definitions of a container family we often omit "treedepth-d" if the constant d is clear from the context.

Neatness is crucial for the following step.

Lemma 5.5. *Let $d > 0$ be fixed. Let G be a graph. Let $F \subseteq D \subseteq V(G)$. Let $(\mathcal{T}, X, \mathsf{Sol})$ be a partial solution in G and let $(\mathcal{T}', X', \mathsf{Sol}')$ be a treedepth-d structure in G that is a neat extension of $\mathcal{T}$ with $V(\mathcal{T}') \setminus V(\mathcal{T}) \subseteq D$.*

Let $\mathcal{T}_F$ consist of $\mathcal{T}$ and ancestors of all nodes of $\mathcal{T}'$ in $D \cap N(F)$. Let $\mathcal{T}'_F$ constist of $\mathcal{T}$ and ancestors of all nodes of $\mathcal{T}'$ in $F \cup (D \cap N(F))$. Then, $(\mathcal{T}'_F, X' \cap V(\mathcal{T}'_F), \mathsf{Sol}' \cap V(\mathcal{T}'_F))$ is a neat extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$, and $V(\mathcal{T}'_F) \setminus V(\mathcal{T}_F) \subseteq F$.

Furthermore, if $\mathcal{A}$ is a threshold automaton, ξ is the multistate assignment that $(\mathcal{T}', X', \mathsf{Sol}')$ evaluates to in $(\mathcal{T}, X, \mathsf{Sol})$ and ξ_F is the multistate assignment that $(\mathcal{T}'_F, X' \cap V(\mathcal{T}'_F), \mathsf{Sol}' \cap V(\mathcal{T}'_F))$ evaluates to in $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$, then if $(\mathcal{T}', X', \mathsf{Sol}')$ is the $\preceq$-minimum extension of $(\mathcal{T}, X, \mathsf{Sol})$ among extensions that evaluate to ξ and satisfy $V(\mathcal{T}') \setminus V(\mathcal{T}) \subseteq D$, then $(\mathcal{T}'_F, X' \cap V(\mathcal{T}'_F), \mathsf{Sol}' \cap V(\mathcal{T}'_F))$ is the $\preceq$-minimum extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$ among extensions that evaluate to ξ_F and satisfy $V(\mathcal{T}'_F) \setminus V(\mathcal{T}_F) \subseteq F$.

Proof. Let $v \in V(\mathcal{T}')$ and $u \in V(\mathcal{T}')$ be such that $u, v \notin V(\mathcal{T})$, $u, v \notin N(F) \cap D$, and u is the parent of v in $\mathcal{T}'$. We claim if exactly one of u, v is in F, then $u, v \in V(\mathcal{T}_F)$.

Since $(\mathcal{T}', X', \mathsf{Sol}')$ neatly extends $(\mathcal{T}, X, \mathsf{Sol})$, there exists a path P from v to u with internal vertices being the descendants of v in $\mathcal{T}'$ (so, in particular, P is disjoint with $\mathcal{T}$). If exactly one of u, v is in F, then P needs to contain an edge xy with $x \in F$ and $y \notin F$. As P is disjoint with $\mathcal{T}$, $V(P) \subseteq D$. Hence, $y \in N(F) \cap D$. Since y is a descendant of v in $\mathcal{T}'$, we have $u, v \in V(\mathcal{T}_F)$, as desired.

We infer that if $v \in V(\mathcal{T}'_F) \setminus V(\mathcal{T}_F)$, then all descendants of v in $\mathcal{T}'$ are also in $V(\mathcal{T}'_F) \setminus V(\mathcal{T}_F)$. Also, if additionally $v \in F$, then all descencendants of v in $\mathcal{T}'$ are in F. Hence, $(\mathcal{T}'_F, X' \cap V(\mathcal{T}'_F), \mathsf{Sol}' \cap V(\mathcal{T}'_F))$ is an extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$ and $V(\mathcal{T}'_F) \setminus V(\mathcal{T}_F) \subseteq F$. Neatness is immediate from the definition.

For the last claim, assume that $(\mathcal{T}''_F, X'', \mathsf{Sol}'')$ is a $\preceq$-smaller extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$ that evaluates to ξ_F and with $V(\mathcal{T}''_F) \setminus V(\mathcal{T}_F) \subseteq F$. Then, one easily observes that if we replace $\mathcal{T}'_F \setminus \mathcal{T}_F$ with $\mathcal{T}''_F \setminus \mathcal{T}_F$ and similarly for X' and Sol' in $(\mathcal{T}', X', \mathsf{Sol}')$ we obtain an extension of $(\mathcal{T}, X, \mathsf{Sol})$ that evaluates to the same ξ but that is $\preceq$-smaller, a contradiction. ☐

We now prove a lemma that handles components D as in the last case "$D \in \mathcal{D}$" of Theorem 3.1, using Theorem 4.1 and recursion.

Lemma 5.6. *Let $d, \widehat{d} \geq 1$ be fixed integers.*

Let G be a graph, $D \subseteq V(G)$, and $D = L \uplus R$ be a partition, let $\mathcal{D}$ be the family of connected components of $G[L]$ and the connected components of $G[R]$. Assume that every $F \in \mathcal{D}$ is a module of $G[D]$ and let H be the quotient graph of $G[D]/\mathcal{D}$.

Furthermore, assume we are given a defect-d_1 exact decomposition family $\mathcal{F}_H$ of H and, for every $F \in \mathcal{D}$, we are given a defect-d_2 extension-$(d^2 + \widehat{d})$ decomposition family $\mathcal{F}_F$ of $G[D']$.

Then, there exists a defect-$((d+1)d_1 + d_2)$ extension-$\widehat{d}$ decomposition family $\mathcal{F}$ of D in G of size at most $(1 + |\mathcal{F}_H|)^{d_1} + \sum_{F \in \mathcal{D}} |\mathcal{F}_F|$. Furthermore, it can be computed in time polynomial in the input and output size, given G, D, $L \uplus R$, $\mathcal{F}_H$, and $(\mathcal{F}_F)_{F \in \mathcal{D}}$.

Moreover, if all families $\mathcal{F}_F$ for $F \in \mathcal{D}$ are exact, then $\mathcal{F}$ is exact, too.

Proof. Fix $(\mathcal{T}, X, \mathsf{Sol})$, ξ, and $(\mathcal{T}', X', \mathsf{Sol}')$ as in the definition of a extension-$\widehat{d}$ decomposition family of D in G. As in the definition, assume that $(\mathcal{T}', X', \mathsf{Sol}')$ is a neat extension of $(\mathcal{T}, X, \mathsf{Sol})$.

Observe that $\mathcal{D}$ is a partition of D. Let

$$\mathcal{D}_{\mathcal{T}'} = \{F \in \mathcal{D} \mid F \cap \mathcal{T}' \neq \emptyset\}.$$

As $H[\mathcal{D}_{\mathcal{T}'}]$ is isomorphic to an induced subgraph of $\mathcal{T}'$, it is of treedepth at most d. Consequently, by the promise of $\mathcal{F}_H$, there exists a tree decomposition (T_H, β_H) of H such that for every $t \in V(T_H)$ we have $\beta_H(t) \in \mathcal{F}_H$ and $|\beta_H(t) \cap \mathcal{D}_H| \leq d_1$.

Let

$$\mathcal{D}_{\mathcal{T}'}^{\mathrm{big}} = \{F \in \mathcal{D} \mid |F \cap \mathcal{T}'| > d\} \subseteq \mathcal{D}_{\mathcal{T}'}.$$

As $\mathcal{T}'$ cannot contain $K_{d+1,d+1}$, we have the following.

Claim 5.7. *For every $F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$, $\bigcup_{F' \in N_H(F)} F'$ contains at most d elements of $\mathcal{T}$. In particular, $\mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$ is an independent set in H.*

Observe that (T_H, β') where $\beta'(t) = \bigcup_{F \in \beta_H(t)} F$ is a tree decomposition of $G[D]$. However, $\beta'(t)$ may contain unbounded number of elements of $\mathcal{T}'$ if $\beta_H(t)$ contains an element of $\mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$. We now modify (T_H, β_H) to isolate elements of $\mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$, so that we can use the families $\mathcal{F}_F$ to handle them.

As a first step, for every $F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$, we drag all nodes of $N_H(F)$ to be present in every bag where F is present, too. More precisely, we define $\beta_H^1 : V(T_H) \to 2^{V(H)}$ as

$$\beta_H^1(t) := \beta_H(t) \cup \bigcup_{F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}} \cap \beta_H(t)} N_H(F).$$

To see that (T_H, β_H^1) is a tree decomposition of H, note that for every $F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$ and $F' \in N_H(F)$, there exists a node t with $F, F' \in \beta(t)$, and thus $\{t \in V(T_H) \mid F' \in \beta_H^1(t)\}$ is connected. Furthermore, as $\bigcup_{F' \in N_H(F)} F'$ contains at most d elements of $\mathcal{T}$ due to Claim 5.7 and every bag $\beta_H(t)$ contains at most d_1 elements of $\mathcal{D}_{\mathcal{T}'}$, we have

$$\forall_{t \in V(T_H)} \left| \bigcup_{F \in \beta_H^1(t) \setminus \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}} F \cap V(\mathcal{T}') \right| \leq (d+1)d_1.$$

As a second step, for every $F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$, we fix one node $t'_F \in V(T_H)$ with $F \in \beta_H^1(t'_F)$ and add a new degree-1 node t_F to T_H, adjacent to t'_F. Let T_H^2 be the resulting tree. For $t \in V(T_H^2)$, define

$$\beta_H^2(t) := \begin{cases} \beta_H^1(t) \setminus \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}} & \text{if } t \in V(T_H) \\ \beta_H^1(t'_F) & \text{if } t = t_F \text{ for some } F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}. \end{cases}$$

It is immediate that (T_H^2, β_H^2) is a tree decomposition of H and for every $t \in V(T_H^2)$

$$\beta_H^2(t) \cap \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}} = \begin{cases} \emptyset & \text{if } t \in V(T_H) \\ \{F\} & \text{if } t = t_F \text{ for some } F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}. \end{cases}$$

Furthermore, as every bag of (T_H^2, β_H^2) is a subset of some bag of (T_H, β_H^1), we have

$$\forall_{t \in V(T_H^2)} \left| \bigcup_{F \in \beta_H^2(t) \setminus \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}} F \cap V(\mathcal{T}') \right| \leq (d+1)d_1. \tag{2}$$

Fix $F \in \mathcal{D}_{\mathcal{T}'}^{\mathrm{big}}$. Let $\mathcal{T}_F$ consist of $\mathcal{T}$ and ancestors of all nodes of $\mathcal{T}'$ in $D \cap N(F)$; note that $|V(\mathcal{T}_F)| \leq d^2 + \widehat{d}$ as $|D \cap N(F) \cap V(\mathcal{T}')| \leq d$. Let $\mathcal{T}_F'$ constist of $\mathcal{T}$ and ancestors of all nodes of $\mathcal{T}'$ in $F \cup (D \cap N(F))$. Lemma 5.5 asserts that $(\mathcal{T}_F', X' \cap V(\mathcal{T}_F'), \mathsf{Sol}' \cap V(\mathcal{T}_F'))$ is a neat extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$, and $V(\mathcal{T}_F') \setminus V(\mathcal{T}_F) \subseteq F$.

Let ξ_F be the multistate assignment such that $(\mathcal{T}_F', X' \cap V(\mathcal{T}_F'), \mathsf{Sol}' \cap V(\mathcal{T}_F'))$ as an extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$ evaluates to. Lemma 5.5 asserts also that $(\mathcal{T}_F', X' \cap V(\mathcal{T}_F'), \mathsf{Sol}' \cap V(\mathcal{T}_F'))$ is the $\preceq$-minimum extension of $(\mathcal{T}_F, X' \cap V(\mathcal{T}_F), \mathsf{Sol}' \cap V(\mathcal{T}_F))$ among extensions that evaluate fo ξ_F and satisfy $V(\mathcal{T}_F') \setminus V(\mathcal{T}_F) \subseteq F$.

Therefore, the promise for $\mathcal{F}_F$ applies. That is, there exists a tree decomposition (T_F, β_F) for $(\mathcal{T}_F', X' \cap V(\mathcal{T}_F'), \mathsf{Sol}' \cap V(\mathcal{T}_F'))$: for every $t \in V(T_D)$ there exists $A \in \mathcal{F}_F$ such that (1) $\beta_F(t) \subseteq A$, and (2) $|A \cap \mathcal{T}_F'| \leq d_2$. By the definition of $\mathcal{T}_F'$, the last point also implies $|A \cap \mathcal{T}'| \leq d_2$.

We now construct a tree decomposition (T, β) of $G[D]$ as follows. To construct T, start with T_H^2 and, for every $F \in \mathcal{D}_{\mathcal{T}}^{\mathrm{big}}$, replace t_F with T_F, i.e., remove t_F and put T_F instead, with its root adjacent to t_F'. Define β as follows:

$$\beta(t) := \begin{cases} \bigcup_{F \in \beta_H^2(t)} F & \text{if } t \in V(T_H) \\ \left(\bigcup_{F' \in \beta_H^2(t_F')} F' \right) \cup \beta_F(t) & \text{if } t \in V(T_F) \text{ for some } F \in \mathcal{D}_{\mathcal{T}}^{\mathrm{big}}. \end{cases}$$

A direct check shows that (T, β) is a tree decomposition of $G[D]$.

We output the family $\mathcal{F}$ consisting of the following sets:

- For every $A \in \mathcal{F}$ and $Z \subseteq A$ that is independent in H and is of size at most d_1, the set

  $$\left(\bigcup_{F \in A \setminus Z} F \right) \cup \left(\bigcup_{F \in Z} \bigcup_{F' \in N_H(F)} F' \right).$$

  Note that every bag $\beta(t)$ for $t \in V(T_H)$ is of the above form and there are at most $(1 + |\mathcal{F}_H|)^{d_1}$ such sets. Furthermore, by (2), every such set contains at most $(d+1)d_1$ elements of $\mathcal{T}'$.

- For every $F \in \mathcal{D}$ and $B \in \mathcal{F}_F$, the set

  $$\left(\bigcup_{F' \in N_H(F)} F' \right) \cup B.$$

  Note that every bag $\beta(t)$ for $t \in V(T_F)$ for some $F \in \mathcal{D}_{\mathcal{T}}^{\mathrm{big}}$ is contained in a set of the above form (or exactly equal to a set of the above form if the families $\mathcal{F}_F$ are exact), there are at most $\sum_{F \in \mathcal{D}} |\mathcal{F}_F|$ such sets, and every such set contains at most $d + d_2$ elements of $\mathcal{T}'$.

This finishes the proof of the lemma. □

We are now ready to state the technical statement that we will prove by induction on the clique number k.

Theorem 5.8. *For every fixed integers $d, \widehat{d}, k > 0$ and a threshold automaton $\mathcal{A}$, there exists a constant $\widetilde{d}$ and an algorithm that takes on input a vertex-weighted P_7-free graph G and a set $D \subseteq V(G)$ with $\omega(G[D]) \leq k$, runs in polynomial time and computes an family $\mathcal{F}$ of subsets of D that is a defect-$\widetilde{d}$ extension-$\widehat{d}$ treedepth-d container family for D.*

Proof. Let $d, \widehat{d}, k > 0$ be fixed. We will proceed by induction on k. In the base case, $k = 1$, so $G[D]$ is edgeless and $\mathcal{F} = \{\{v\} \mid v \in D\}$ satisfies the requiments with $\widetilde{d} = 1$. Assume then $k > 1$.

Let d_1 be the bound on $|K \cap V(\mathcal{T})|$ of Theorem 3.1 for the fixed d, k. Let $\widetilde{d'}$ be the constant from the inductive assumption for treedepth d, clique number $k-1$, and extension $\widehat{d} + dd_1 + d^2$.

We invoke Theorem 3.1 on $G[D]$, obtaining a family $\mathcal{F}'$. For every $\Psi := (K, \mathcal{D}, L) \in \mathcal{F}'$ and every $D' \in \mathcal{D}$ proceed as follows. Let $\mathcal{C}(D')$ be the family of connected components of either $G[L(D')]$ or of $G[D' \setminus L(D')]$. For every $F \in \mathcal{C}(D')$, recurse with the same treedepth d, clique number $k-1$, and extension $\widehat{d} + dd_1 + d^2$, obtaining a defect-$\widetilde{d'}$ extension-$(\widehat{d} + dd_1 + d^2)$ treedepth-d family $\mathcal{F}_F$. Invoke Theorem 4.1 on the quotient graph $G[\bigcup \mathcal{D}]/\bigcup_{D' \in \mathcal{D}} \mathcal{C}(D')$, obtaining a family $\mathcal{F}^{\mathrm{bip}}$. Invoke Lemma 5.6 in G, $\bigcup \mathcal{D}$, partition $(\bigcup_{D' \in \mathcal{D}} L(D')) \uplus (\bigcup_{D' \in \mathcal{D}} D' \setminus L(D'))$, the family $\mathcal{F}^{\mathrm{bip}}$ and families $(\mathcal{F}_F)_{D' \in \mathcal{D}, F \in \mathcal{C}(D')}$, obtaining a family $\mathcal{F}''$. Note that $\mathcal{F}''$ is a defect-$\widetilde{d''}$ extension-$(\widehat{d} + dd_1)$ treedepth-d container family for $\bigcup \mathcal{D}$ in G for some constant $\widetilde{d''}$.

Construct a family $\mathcal{F}^{\mathrm{sep}}$ as follows. Iterate over all partial solutions $(\mathcal{T}_\Psi, X_\Psi, \mathsf{Sol}_\Psi)$ in G with $|V(\mathcal{T}_\Psi)| \leq \widehat{d} + dd_1$ and their multistate assignments ξ_Ψ. Set $\mathcal{F}_{\Psi,\mathcal{T}} = \{V(\mathcal{T}_\Psi) \cup A \mid A \in \mathcal{F}''$. For every choice of $(\mathcal{T}_\Psi, X_\Psi, \mathsf{Sol}_\Psi)$ and ξ_Ψ, use $\mathcal{F}_{\Psi,\mathcal{T}}$ and the algorithm of [13, Section 3] (Lemma 5.3) to compute in polynomial time the $\preceq$-minimum extension $(\mathcal{T}'_\Psi, X'_\Psi, \mathsf{Sol}'_\Psi)$ of $(\mathcal{T}_\Psi, X_\Psi, \mathsf{Sol}_\Psi)$ among extensions that evaluate to ξ and satisfy $V(\mathcal{T}'_\Psi) \setminus V(\mathcal{T}_\Psi) \subseteq \bigcup \mathcal{D}$. Insert $K \cup (\bigcup \mathcal{D} \setminus V(\mathcal{T}'_\Psi))$ into $\mathcal{F}^{\mathrm{sep}}$.

Let $(\mathcal{T}, X, \mathsf{Sol})$ be a partial solution with $|V(\mathcal{T})| \leq \widehat{d}$, let ξ be a multistage assignment of $(\mathcal{T}, X, \mathsf{Sol})$, and let $(\mathcal{T}', X', \mathsf{Sol}')$ be the $\preceq$-minimum extension of $(\mathcal{T}, X, \mathsf{Sol})$ amongs ones that evaluate to ξ and satisfy $V(\mathcal{T}') \setminus V(\mathcal{T}) \subseteq D$. Assume $(\mathcal{T}', X', \mathsf{Sol}')$ is a neat extension of $(\mathcal{T}, X, \mathsf{Sol})$.

Let $\mathcal{T}''$ be any maximal treedepth-d structure that contains $V(\mathcal{T}') \cap D$. Let F'' be any $\mathcal{T}''$-aligned minimal chordal completion of $G[D]$. Let S be a $\mathcal{T}''$-avoiding minimal separator of $G[D] + F''$ with full components A, B. Let $\Psi = (K, \mathcal{D}, L) \in \mathcal{F}'$ be the tuple for S, A, B promised by Theorem 3.1. Set $F := \bigcup \mathcal{D}$ and use Lemma 5.5 to G, D, F, $(\mathcal{T}, X, \mathsf{Sol})$ and $(\mathcal{T}', X', \mathsf{Sol}')$. We infer that $\mathcal{F}^{\mathrm{sep}}$ contains $K \cup (F \setminus V(\mathcal{T}'))$. As K is guaranteed to contain $V(\mathcal{T}') \cap S$, we have that $S \subseteq K \cup (F \setminus V(\mathcal{T}'))$. That is, $\mathcal{F}^{\mathrm{sep}}$ contains a set that on one hand contains S, and on the other hand contains at most $d_1 + \widetilde{d''}$ elements of $\mathcal{T}''$.

Let $c = c(k)$ be the constant from Corollary 2.13 for $t = 7$ and k. We set $\mathcal{F}$ to be the union of $\{N[v] \mid v \in V(G)\}$ and the set of all possible unions at most c elements of $\mathcal{F}^{\mathrm{sep}}$. Corollary 2.13 implies that for every maximal clique Ω of $G[D] + F''$ there exists $A \in \mathcal{F}$ with $\Omega \subseteq A$ and $|A \cap V(\mathcal{T}')| \leq c(d_1 + \widetilde{d''})$. Hence, a clique tree (T, β) of $G[D] + F''$ satisfies the requirements of the theorem for $\widetilde{d} = c(d_1 + \widetilde{d''})$. This finishes the proof. $\square$

Theorem 1.1 follows now from the combination of Theorem 5.8 for $\widehat{d} = 0$ and $D = V(G)$ and the algorithm of [13, Section 3] (Lemma 5.3). Note that for the case $\widehat{d}$, any $\preceq$-minimum extension $(\mathcal{T}', X', \mathsf{Sol}')$ of $(\emptyset, \emptyset, \emptyset)$ among extensions that evaluate to some fixed multistate assignment ξ satisfies that $\mathcal{T}'$ is maximal and neat, as proven in [13].

6 Conclusion

Our work suggests some directions for future research. Of course the most ambitious goal is to show a polynomial-time algorithm for $(\mathrm{tw} \leq d, \psi)$-MWIS in P_t-free graphs, for all fixed t, with the first open case for $t = 7$ (with no further assumptions on the graph).

However, there are several intermediate goals that also seem interesting. For example, we think that the idea of considering graphs of bounded clique number is quite promising. Not only it allows us to use some strong structural tools like χ-boundedness or VC-dimension, but also to measure the progress of an algorithm by decreasing clique number (see also [12, 11]).

Problem 6.1 ([12, Problem 9.2])**.** *Show that for every fixed t and k, MWIS is polynomial-time solvable in P_t-free graphs with clique number at most k.*

As illustrated in Section 4, assuming additionally that a graph is bipartite might lead to an easier, but still interesting problem. Of course in this setting asking for an algorithm for MWIS makes little sense, but already the next case is far from trivial.

Problem 6.2. *Show that for every fixed t, given a vertex-weighted bipartite P_t-free graph, in polynomial-time we can find an induced forest of maximum possible weight.*

Recall that the polynomial-time algorithm for $t = 7$ follows not only from our work, but also (via a very different approach) from the work of Lozin and Zamaraev [33] – their structural theorem about P_7-free bipartite graphs have bounded mim-width. However, this is no longer the case for $P - 8$-free bipartite graphs, as shown by Brettell et al. [10].

A different subclass of P_t-free graphs are $(P_t, K_{1,k})$-free graphs (i.e., excluding additionally a star with k leaves as an induced subgraph), studied by Lozin and Rautenbach [32], who show that MAXIMUM WEIGHT INDEPENDENT SET is polynomial-time solvable in these graph classes. Does this result generalize to $(\mathrm{tw} \leq d, \psi)$-MWIS?

Finally, let us point out that Abrishami, Chudnovsky, Pilipczuk, Rzążewski, and Seymour [1] did not only provide an algorithm for $(\mathrm{tw} \leq d, \psi)$-MWIS in P_5-free graphs, but they actually considered a richer class of graphs. In particular, their algorithm works for $C_{>4}$-free graphs, i.e., graph with no induced cycles of length more than 4 (analogously we define $C_{>t}$-free graphs for any t). Similarly, the quasipolynomial-time algorithm for $(\mathrm{tw} \leq d, \psi)$-MWIS works for $C_{>t}$-free graphs, for any t. Note that $C_{>t}$-free graphs are a proper superclass of P_t-free graphs.

We believe that all polynomial-time results for P_t-free graphs, discussed in this paper, can be actually lifted to $C_{>t}$-free graphs.

References

[1] Tara Abrishami, Maria Chudnovsky, Marcin Pilipczuk, Paweł Rzążewski, and Paul D. Seymour. Induced subgraphs of bounded treewidth and the container method. *SIAM J. Comput.*, 53(3):624–647, 2024.

[2] Benjamin Bergougnoux, Édouard Bonnet, Nick Brettell, and O-joung Kwon. Close relatives of feedback vertex set without single-exponential algorithms parameterized by treewidth. In Yixin Cao and Marcin Pilipczuk, editors, *15th International Symposium on Parameterized and Exact Computation, IPEC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, volume 180 of *LIPIcs*, pages 3:1–3:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[3] Benjamin Bergougnoux, Jan Dreier, and Lars Jaffke. A logic-based algorithmic meta-theorem for mim-width. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 3282–3304. SIAM, 2023.

[4] Marthe Bonamy, Nicolas Bousquet, Michał Pilipczuk, Paweł Rzążewski, Stéphan Thomassé, and Bartosz Walczak. Degeneracy of P_t-free and $C_{>t}$-free graphs with no large complete bipartite subgraphs. *J. Comb. Theory B*, 152:353–378, 2022.

[5] Flavia Bonomo, Maria Chudnovsky, Peter Maceli, Oliver Schaudt, Maya Stein, and Mingxian Zhong. Three-coloring and list three-coloring of graphs without induced paths on seven vertices. *Comb.*, 38(4):779–801, 2018.

[6] Flavia Bonomo-Braberman, Maria Chudnovsky, Jan Goedgebeur, Peter Maceli, Oliver Schaudt, Maya Stein, and Mingxian Zhong. Better 3-coloring algorithms: Excluding a triangle and a seven vertex path. *Theor. Comput. Sci.*, 850:98–115, 2021.

[7] Vincent Bouchitté and Ioan Todinca. Listing all potential maximal cliques of a graph. *Theor. Comput. Sci.*, 276(1-2):17–32, 2002.

[8] Vincent Bouchitté and Ioan Todinca. Treewidth and minimum fill-in: Grouping the minimal separators. *SIAM J. Comput.*, 31(1):212–232, jan 2002.

[9] Andreas Brandstädt and Raffaele Mosca. Maximum Weight Independent Sets for (P_7, triangle)-free graphs in polynomial time. *Discret. Appl. Math.*, 236:57–65, 2018.

[10] Nick Brettell, Jake Horsfield, Andrea Munaro, Giacomo Paesani, and Daniël Paulusma. Bounding the mim-width of hereditary graph classes. *J. Graph Theory*, 99(1):117–151, 2022.

[11] Nick Brettell, Jake Horsfield, Andrea Munaro, and Daniël Paulusma. List k-colouring P_t-free graphs: A mim-width perspective. *Inf. Process. Lett.*, 173:106168, 2022.

[12] Maria Chudnovsky, Jason King, Michał Pilipczuk, Paweł Rzążewski, and Sophie Spirkl. Finding large H-colorable subgraphs in hereditary graph classes. *SIAM J. Discret. Math.*, 35(4):2357–2386, 2021.

[13] Maria Chudnovsky, Rose McCarty, Marcin Pilipczuk, Michał Pilipczuk, and Paweł Rzążewski. Sparse induced subgraphs in P_6-free graphs. In David P. Woodruff, editor, *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024, Alexandria, VA, USA, January 7-10, 2024*, pages 5291–5299. SIAM, 2024.

[14] Derek G. Corneil, Yehoshua Perl, and Lorna K. Stewart. A linear recognition algorithm for cographs. *SIAM J. Comput.*, 14(4):926–934, 1985.

[15] Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique width. In Juraj Hromkovic and Ondrej Sýkora, editors, *Graph-Theoretic Concepts in Computer Science, 24th International Workshop, WG '98, Smolenice Castle, Slovak Republic, June 18-20, 1998, Proceedings*, volume 1517 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 1998.

[16] Clément Dallard, Martin Milanič, and Kenny Storgel. Treewidth versus clique number. II. tree-independence number. *J. Comb. Theory B*, 164:404–442, 2024.

[17] Guo Li Ding, Paul Seymour, and Peter Winkler. Bounding the vertex cover number of a hypergraph. *Combinatorica*, 14(1):23–34, March 1994.

[18] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.

[19] Fedor V. Fomin, Ioan Todinca, and Yngve Villanger. Large induced subgraphs via triangulations and CMSO. *SIAM J. Comput.*, 44(1):54–87, 2015.

[20] Peter Gartland and Daniel Lokshtanov. Independent set on P_k-free graphs in quasi-polynomial time. In Sandy Irani, editor, *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 613–624. IEEE, 2020.

[21] Peter Gartland, Daniel Lokshtanov, Tomás Masařík, Marcin Pilipczuk, Michał Pilipczuk, and Paweł Rzążewski. Maximum weight independent set in graphs with no long claws in quasi-polynomial time. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*, pages 683–691. ACM, 2024.

[22] Peter Gartland, Daniel Lokshtanov, Marcin Pilipczuk, Michał Pilipczuk, and Paweł Rzążewski. Finding large induced sparse subgraphs in $C_{>t}$-free graphs in quasipolynomial time. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 330–341. ACM, 2021.

[23] Andrzej Grzesik, Tereza Klimošová, Marcin Pilipczuk, and Michał Pilipczuk. Polynomial-time algorithm for maximum weight independent set on P_6-free graphs. *ACM Trans. Algorithms*, 18(1):4:1–4:57, 2022.

[24] Martin Grötschel, Laszlo Lovász, and Alexander Schrijver. Polynomial algorithms for perfect graphs. In C. Berge and V. Chvátal, editors, *Topics on Perfect Graphs*, volume 88 of *North-Holland Mathematics Studies*, pages 325–356. North-Holland, 1984.

[25] András Gyárfás. Problems from the world surrounding perfect graphs. *Applicationes Mathematicae*, 19(3-4):413–441, 1987.

[26] Frédéric Havet, Ross J. Kang, and Jean-Sébastien Sereni. Improper coloring of unit disk graphs. *Networks*, 54(3):150–164, 2009.

[27] Bart M. P. Jansen, Daniel Lokshtanov, and Saket Saurabh. A near-optimal planarization algorithm. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1802–1811. SIAM, 2014.

[28] Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger, editors, *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20–22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, and sponsored by the Office of Naval Research, Mathematics Program, IBM World Trade Corporation, and the IBM Research Mathematical Sciences Department*, pages 85–103, Boston, MA, 1972. Springer US.

[29] Ton Kloks, Ching-Hao Liu, and Sheung-Hung Poon. Feedback vertex set on chordal bipartite graphs. *CoRR*, abs/1104.3915, 2012.

[30] Paloma T. Lima, Martin Milanič, Peter Mursic, Karolina Okrasa, Pawel Rzążewski, and Kenny Štorgel. Tree decompositions meet induced matchings: beyond max weight independent set. *CoRR*, abs/2402.15834, 2024.

[31] Daniel Lokshtanov, Martin Vatshelle, and Yngve Villanger. Independent set in P_5-free graphs in polynomial time. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 570–581. SIAM, 2014.

[32] Vadim V. Lozin and Dieter Rautenbach. Some results on graphs without long induced paths. *Inf. Process. Lett.*, 88(4):167–171, 2003.

[33] Vadim V. Lozin and Viktor Zamaraev. The structure and the number of P_7-free bipartite graphs. *Eur. J. Comb.*, 65:143–153, 2017.

[34] Pranabendu Misra, Venkatesh Raman, M. S. Ramanujan, and Saket Saurabh. Parameterized algorithms for even cycle transversal. In Martin Charles Golumbic, Michal Stern, Avivit Levy, and Gila Morgenstern, editors, *Graph-Theoretic Concepts in Computer Science - 38th International Workshop, WG 2012, Jerusalem, Israel, June 26-28, 2012, Revised Selcted Papers*, volume 7551 of *Lecture Notes in Computer Science*, pages 172–183. Springer, 2012.

[35] Giacomo Paesani, Daniël Paulusma, and Paweł Rzążewski. Feedback Vertex Set and Even Cycle Transversal for H-free graphs: Finding large block graphs. *SIAM J. Discret. Math.*, 36(4):2453–2472, 2022.

[36] Marcin Pilipczuk. A tight lower bound for Vertex Planarization on graphs of bounded treewidth. *Discret. Appl. Math.*, 231:211–216, 2017.

[37] Marcin Pilipczuk, Michał Pilipczuk, and Paweł Rzążewski. Quasi-polynomial-time algorithm for independent set in P_t-free graphs via shrinking the space of induced paths. In Hung Viet Le and Valerie King, editors, *4th Symposium on Simplicity in Algorithms, SOSA 2021, Virtual Conference, January 11-12, 2021*, pages 204–209. SIAM, 2021.

[38] Marcin Pilipczuk and Paweł Rzążewski. A polynomial bound on the number of minimal separators and potential maximal cliques inP_6-free graphs of bounded clique number. *CoRR*, abs/2310.11573, 2023.

[39] Vladimir N. Vapnik and Alexei Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.