Fast fixed-parameter tractable algorithms for nontrivial generalizations of vertex cover *

Naomi Nishimura[†] Prabhakar Ragde[†] Dimitrios M. Thilikos [‡]

Keywords: graph minors, obstructions, vertex cover

Abstract

Our goal in this paper is the development of fast algorithms for recognizing general classes of graphs. We seek algorithms whose complexity can be expressed as a linear function of the graph size plus an exponential function of k, a natural parameter describing the class. In particular, we consider the class $\mathcal{W}_k(\mathcal{G})$, where for each graph G in $\mathcal{W}_k(\mathcal{G})$, the removal of a set of at most k vertices from G results in a graph in the base graph class $\mathcal{G}$. (If $\mathcal{G}$ is the class of edgeless graphs, $\mathcal{W}_k(\mathcal{G})$ is the class of graphs with bounded vertex cover.)

When $\mathcal{G}$ is a minor-closed class such that each graph in $\mathcal{G}$ has bounded maximum degree, and all obstructions of $\mathcal{G}$ (minor-minimal graphs outside $\mathcal{G}$) are connected, we obtain an $O((g+k)|V(G)| + (fk)^k)$ recognition algorithm for $\mathcal{W}_k(\mathcal{G})$, where g and f are constants (modest and quantified) depending on the class $\mathcal{G}$. If $\mathcal{G}$ is the class of graphs with maximum degree bounded by D (not closed under minors), we can still obtain a running time of $O(|V(G)|(D+k) + k(D+k)^{k+3})$ for recognition of graphs in $\mathcal{W}_k(\mathcal{G})$.

Our results are obtained by considering minor-closed classes for which all obstructions are connected graphs, and showing that the size of any obstruction for $\mathcal{W}_k(\mathcal{G})$ is $O(tk^7 + t^7k^2)$, where t is a bound on the size of obstructions for $\mathcal{G}$. A trivial corollary of this result is an upper bound of $(k+1)(k+2)$ on the number of vertices in any obstruction of the class of graphs with vertex cover of size at most k. These results are of independent graph-theoretic interest.

1 Introduction

One of the principal goals of algorithmic graph theory is to determine elegant and efficient ways to characterize classes of graphs. A particularly enticing approach applies readily to graph classes

*Reserach supported by the Natural Sciences and Engineering Research Council of Canada, Ministry of Education and Culture of Spain (grant number MEC-DGES SB98 0K148809), and EU project ALCOM-FT (IST-99-14186).

[†]Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1.

[‡]Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Campus Nord, Mòdul C5, c/Jordi Girona Salgado, 1-3. E-08034, Barcelona, Spain.

closed under minor containment (defined formally in Section 2). In their seminal work on graph minors, Robertson and Seymour [RS85b] proved that for any minor-closed graph class $\mathcal{G}$, there is a finite number of minor-minimal graphs (the set of *obstructions*, or *obstruction set*) in the set of graphs outside of $\mathcal{G}$. As a consequence, G is in $\mathcal{G}$ if and only if no graph in the obstruction set of $\mathcal{G}$ is a minor of G; if the obstruction set of $\mathcal{G}$ is known, there exists a polynomial-time recognition algorithm for $\mathcal{G}$ [RS85a, RS95].

Unfortunately, finding the obstruction set of a class $\mathcal{G}$ is unsolvable in general [FRS87, FL94, van90] and appears to be a hard structural problem even for simple graph classes, largely due to the rapid explosion in the size of the obstructions [TUK94, Ram95, Thi00]. Following a brute force approach, it is possible to build a computer program enumerating graphs and searching among them for obstructions. A crucial drawback of this method is that there is no general way to bound the search space [van90, FL94]; more sophisticated methods are also possible [Din95]. It is thus necessary to determine upper bounds on the sizes of the obstructions for special graph classes. Results of this type have been obtained for graphs with bounded treewidth or pathwidth [Lag98] as well as more general graphs [Lag93].

In this paper we settle the question of the combinatorial growth for graph classes created from simpler ones. We augment a graph class by adding at most k vertices (and adjacent edges) to each graph in the class. More formally, for each graph class $\mathcal{G}$ and integer $k \geq 1$, we define $\mathcal{W}_k(\mathcal{G})$ to consist of graphs G which are *within k vertices of $\mathcal{G}$*, namely all graphs G such that the vertices of G can be partitioned into sets S_1 and S_2, where $|S_1| \leq k$ and the subgraph of G induced on the vertices in S_2 is in the class $\mathcal{G}$. The notion of "within k" can be used to easily define the classes of graphs with vertex cover at most k and graphs with vertex feedback set at most k ($\mathcal{G}$ is the class of edgeless graphs and the class of forests, respectively). Moreover, if $\mathcal{G}$ is closed under taking of minors, then so is $\mathcal{W}_k(\mathcal{G})$ [FL88, LP98]. We conjecture that we can relate the sizes of obstructions for $\mathcal{G}$ to those for $\mathcal{W}_k(\mathcal{G})$.

Conjecture 1 *If $\mathcal{G}$ is a minor-closed graph class whose obstructions have no more than t vertices, then the size of the obstructions for $\mathcal{W}_k(\mathcal{G})$ depends only on k and t.*

We show that the above conjecture is true for any class $\mathcal{G}$ in which no graph has degree greater than a fixed constant D, provided the class is minor-closed with all obstructions connected (in the proof of Lemma 14, we show that this is equivalent to the class being closed under disjoint union). In particular, we prove that for such a $\mathcal{G}$, any graph in the obstruction set of $\mathcal{W}_k(\mathcal{G})$ has size bounded by a polynomial in k, D, and (for $D \geq 3$) C, where C is an upper bound on the length of paths of degree-two vertices ("induced chains") in the obstruction set of $\mathcal{G}$. As intermediate steps in proving this result, we develop lemmas demonstrating the existence of leafy trees in sparse (bounded degree) graphs, constituting results of independent interest.

Making use of advances in construction of fixed-parameter tractable algorithms [DF99, Nie98], we employ the method of "reduction to a problem kernel" to obtain recognition algorithms such that an exponential function on k contributes only an additive term to the overall complexity. This builds on previous work [Meh84, BG93]; our results can be seen as generalizations of the algorithms designed for the vertex-cover problem [BFR98, NR99, CKJ99]. Fast algorithms of this type have been generated for other problems [DF95, FPT95, KST99, DFS99, CCDF97, MR99]. We make use of our upper bound on the size of obstructions to obtain an $O((g(t)+k)|V(G)| + (f(k,t))^k)$-time algorithm recognizing $\mathcal{W}_k(\mathcal{G})$ when $\mathcal{G}$ satisfies the following three conditions: $\mathcal{G}$ is closed under taking of minors; each graph in $\mathcal{G}$ has degree bounded by D; and the obstructions for $\mathcal{G}$ are connected and of size bounded by t. Here g and f are polynomial functions with multiplicative constants depending on the class $\mathcal{G}$. Finally, we demonstrate that similar time complexities can be achieved even when the closure restrictions are removed. We present an $O(|V(G)|(D+k) + k(D+k)^{k+3})$-time algorithm for the recognition of $\mathcal{W}_k(\mathcal{G})$, where $\mathcal{G}$ is the class of graphs with maximum degree bounded by D.

After establishing notation in Section 2, we demonstrate in Section 3 that we can obtain a bound on the size of a graph G when there are bounds on the degree and length of chains and the disjoint union of stars is excluded as a minor. Building on this result, in Section 4 we establish a bound on the size of obstructions for $\mathcal{W}_k(\mathcal{G})$. Sections 5 and 6 establish polynomial-time algorithms, in turn making use of earlier results.

2 Preliminaries

In discussing graphs, we make use of standard notation in graph theory. For each graph G considered, we will denote as $V(G)$ and $E(G)$ its vertex and edge set respectively. For $\mathcal{G}$ a finite graph class, we define $\text{max-size}(\mathcal{G}) = \max\{|V(G)| \mid G \in \mathcal{G}\}$. If $S \subseteq V(G)$, we define $\mathrm{nbr}_G(S)$ to be the set of vertices outside S that are adjacent to vertices in S, and $G[S]$ to be the subgraph of G induced on S. If $v \in V(G)$, we define the *degree of v in G* to be $\deg_G(v) = |\mathrm{nbr}_G(\{v\})|$, the set of *pendant vertices* to be $\mathrm{pend}(G) = \{v \in V(G) \mid \deg_G(v) = 1\}$, and the set of *internal vertices* to be $V(G) \setminus \mathrm{pend}(G)$. By maximizing $\deg_G(v)$ over all vertices v, we obtain the degree bound on the graph G, denoted $\Delta(G)$. We call a graph class $\mathcal{G}$ *bounded degree* if the quantity $\Delta(\mathcal{G}) = \max\{\Delta(G) \mid G \in \mathcal{G}\}$ can be bounded above by a constant.

At times we will alter a graph by replacing induced paths by edges. For $u, v \in V(G)$, we define $\mathrm{dist}(u,v)$ to be the length of the shortest path connecting u and v (the number of edges in the path). We call a path of a graph G an *a-chain* if it has length a, all its internal vertices have degree 2 in G, and its end vertices are either adjacent or have degree not equal to two. We denote as $\mathrm{chain}(G)$ the largest a for which there exists an a-chain in G, setting $\mathrm{chain}(G) = 0$

when $E(G) = \emptyset$. A graph G is *resolved* if $\text{chain}(G) = 1$. Finally, for any graph class $\mathcal{G}$, we set $\text{chain}(\mathcal{G}) = \max\{\text{chain}(G) \mid G \in \mathcal{G}\}$.

Throughout this paper we will use $\mathcal{G}$ to denote a graph class that is closed under taking of minors or, equivalently, a *minor-closed* graph class. A graph G is a *minor of* a graph H if a graph isomorphic to G can be formed from H by a series of edge and vertex deletions and edge contractions. The *contraction* of an edge $e = (u, v)$ in G results in a graph G', in which u and v are replaced by a new vertex v_e ($V(G') = \{v_e\} \cup V(G) \setminus \{u, v\}$) and in which for every neighbor w of u or v in G, there is an edge (w, v_e) in G'. An *obstruction* of G is a minor-minimal graph outside $\mathcal{G}$; we use $\text{ob}(\mathcal{G})$ to denote the set of obstructions of $\mathcal{G}$, or the *obstruction set* of $\mathcal{G}$. We use $G \preceq H$ and $G \leq H$ to denote that G is a minor of or an induced subgraph of H, respectively.

We use K_r to denote the complete graph on r vertices and $K_{r,s}$ to denote the complete bipartite graph with r and s vertices in the two parts of the bipartition. The graph $K_{1,r}$ is also known as a *star*. We use L_r^{k+1} to denote the graph consisting of $k+1$ disjoint copies of $K_{1,r}$.

For any integer $k \geq 1$, we define a set of graphs that are *within k vertices* of $\mathcal{G}$, parameterized by k, the number of vertices that need to be removed to form a graph in $\mathcal{G}$, namely $\mathcal{W}_k(\mathcal{G}) = \{G \mid \exists S \subseteq V(G) \text{ such that } |S| \leq k \text{ and } G[V(G) - S] \in \mathcal{G}\}$.

3 Excluding disjoint stars

In order to prove a bound on the size of the largest obstruction, we first establish Theorem 1, which shows that the absence of a L_r^{k+1} minor combined with a bound on the degree and the length of a chain results in a graph of bounded size. In Section 4 we will make use of the theorem by showing that any sufficiently large graph with the aforementioned restrictions must contain L_r^{k+1} as a minor.

Theorem 1 *Any L_r^{k+1}-minor-free connected graph G where $\Delta(G) \leq D$ and $\text{chain}(G) \leq C$ has at most $f(k, r, D, C)$ vertices, where for any four integers $k \geq 0, r \geq 1$, $D \geq 3$, and $C \geq 0$, we define $\gamma = \max\{r-1, D\}$ and $f(k, r, D, C)$ as follows:*

$$f(k,r,D,C) = \begin{cases} (k+1)(D+1), & \text{if } r = 1 \\ (D+1)(k(D^2+1)+1), & \text{if } r = 2 \\ \frac{5}{2}(D+1)(k(D^2+1)+1))(CD+2) & \text{if } r = 3 \\ 5(CD+2)(\gamma(1+\gamma)(k(4(\gamma-1)^2+1)+1)-1) & \text{if } r \geq 4 \end{cases}$$

In order to prove Theorem 1, we prove a series of lemmas, each used in proofs by contradiction for a range of values of r. When $r = 1$, the lower bound on the size of a matching established in Lemma 1 (whose proof is elementary [Lov79, exercise 7.25]) implies the presence of L_r^{k+1}.

Lemma 1 *Any connected graph G contains a matching of size $\frac{|V(G)|}{\Delta(G)+1}$.* □

When $r = 2$ or $r = 3$, we use Lemma 2 to obtain a proof by contradiction. In each case, we show that if $|V(G)|$ is large, we are able to obtain a tree minor T of G with a large number of sufficiently high degree vertices, showing that $L_r^{k+1} \preceq T \preceq G$. Lemma 3 is used to count vertices of degree at least two in T.

Lemma 2 *If T is a tree, R is the set of vertices of degree at least r in T, and $|R| > k((\Delta(T))^2 + 1)$, then $L_r^{k+1} \preceq T$.*

Proof. In the pseudocode below, we extract a set I of $k+1$ vertices and show that the vertices in the set are centers of disjoint stars $K_{1,r}$. We repeatedly select an arbitrary high degree vertex, and then eliminate that vertex as well as all high-degree nodes at distance at most two from it.

> $I := \emptyset$, $I^* := \emptyset$
> `while` $|I| \leq k$ `do`
> $v_i :=$ `an arbitrary vertex in` $R - I^*$
> $N_i := \{v \in R \mid \mathrm{dist}_T(v, v_i) \leq 2\}$
> $I := I \cup \{v_i\}$
> $I^* := I^* \cup N_i$
> `output` I

To prove that the output set I contains at least $k+1$ vertices, we must show that after k iterations, $R - I^*$ is nonempty. For any vertex v there is one vertex at distance 0, at most $\Delta(T)$ at distance 1, and at most $(\Delta(T) - 1)\Delta(T)$ at distance 2; thus $|N_i| \leq (\Delta(T))^2 + 1$. As a consequence, the ith time that the procedure enters the loop, $|R - I^*| \geq |R| - (i-1)((\Delta(T))^2 + 1)$. The fact that $|R| > k((\Delta(T))^2 + 1)$ guarantees that $R - I^*$ is nonempty during the first $k+1$ repetitions of the loop, as needed.

It will now suffice to show that each vertex in I is the center of one component of L_r^{k+1}. By construction, at each iteration I^* is augmented by the vertices of distance at most two from the vertices added to I. This guarantees that for any pair $\{v_i, v_j\}$ of vertices in I, $\mathrm{dist}_T(v_i, v_j) \geq 3$. The graphs $T[N_i]$ and $T[N_j]$ are thus disjoint; since $I \subseteq R$, each one of them contains $K_{1,r}$ as a minor, as needed to complete the proof of the lemma. □

Lemma 3 *For any connected graph G, $|V(G) - \mathrm{pend}(G)| \geq \frac{|\mathrm{pend}(G)|}{\Delta(G)}$ and $|V(G) - \mathrm{pend}(G)| \geq \frac{|V(G)|}{\Delta(G)+1}$.*

Proof. We consider the set $N = \mathrm{nbr}_G(\mathrm{pend}(G))$, the neighbors of vertices in $\mathrm{pend}(G)$. By the definition of $\Delta(G)$, $|\mathrm{pend}(G)| \leq \Delta(G)|N|$. Since $N \subseteq V(G) - \mathrm{pend}(G)$, clearly $|\mathrm{pend}(G)| \leq \Delta(G)|V(G) - \mathrm{pend}(G)|$, proving the first part of the lemma. For the second part, we observe that $|V(G)| = |\mathrm{pend}(G)| + |V(G) - \mathrm{pend}(G)| \leq \Delta(G)|V(G) - \mathrm{pend}(G)| + |V(G) - \mathrm{pend}(G)| = |V(G) - \mathrm{pend}(G)|(\Delta(G) + 1)$. □

In order to apply Lemma 2 to the case $r = 3$, we need to find a tree minor in which each internal vertex has degree at least three. Lemmas 4 and 5 are applied sequentially to find successive minors which are a connected resolved graph and a resolved tree.

Lemma 4 *Any connected graph G contains as a minor a connected resolved graph H such that $|V(H)| > \frac{2|V(G)|}{\mathrm{chain}(G)\Delta(G)+2}$ and $\Delta(H) = \Delta(G)$.*

Proof. We let V^2 be the set of vertices that have degree two and have two neighbours of degree two, V^* be the set $V(G) - V^2$, and E^2 be the set of edges of G containing at least one endpoint in V^2. For R the set of chains in G, V^2 is exactly the set of internal vertices of chains in R and E^2 is exactly the set of edges of chains in R.

We wish to show that H, the graph obtained by replacing each chain by an edge between its end vertices, has the necessary properties. We first determine an upper bound on the size of V^*. As any chain in R contains fewer than $\mathrm{chain}(G)$ internal vertices, $|V^2| < \mathrm{chain}(G)|R|$. Moreover, $2|R| \leq \Delta(G)|V^*|$, since each vertex in V^* has at most $\Delta(G)$ neighbors (and hence is an end vertex of at most $\Delta(G)$ chains of R), and any chain in R contains exactly two vertices in V^*, namely its end vertices. Combining these two observations, $|V(G)| - |V^*| = |V^2| < \frac{1}{2}\mathrm{chain}(G)\Delta(G)|V^*|$, or, equivalently, $|V^*| > \frac{2|V(G)|}{\mathrm{chain}(G)\Delta(G)+2}$.

We now construct H from G by replacing each chain by an edge between its end vertices. By construction, H is a minor of G and has the same maximum degree as G. Moreover, the vertices in H are exactly the vertices in V^*, as needed to complete the proof. □

Lemma 5 *Any connected resolved graph G contains as a minor a resolved tree T such that $|V(T)| \geq \frac{|V(G)|}{5}$ and $\Delta(T) \leq \Delta(G)$.*

Proof. We construct T from S, a spanning tree of G with the maximum number of leaves.

We first show that S cannot contain a path (t, u, v, w, x, y, z) such that all internal vertices have degree two. Suppose instead that S has such a path. By definition, $\Delta_G(w) \geq 3$, and therefore there exists an edge $e = (w, r)$ incident to w that is not an edge of S. As S is a spanning tree, there is a cycle in G formed by e and edges in $E(S)$; as u, v, x, y all have degree 3 in S, this cycle contains either t or z, and therefore, without loss of generality, we assume (v, w) is an edge of this cycle. However, we could then form $S' = (V(S), E(S) \cup \{e\} - \{(v, w)\})$, which is a spanning tree of G. It is possible for r to be a nonleaf vertex in S' and a leaf vertex in S.

However, u and v are both leaves in S' but not in S, and hence S' has at least one more leaf than S, contradicting the maximality of S.

We now construct T from S by replacing each a-chain in S, $a \geq 2$, by an edge. As we have seen that no chain has more than six vertices, T has at least $|E(S)|/5$ edges and thus, $|V(T)| \geq \frac{|V(S)|-1}{5} + 1 \geq \frac{|V(S)|}{5}$. By its construction, T is a minor of G that does not have vertices of degree 2 and such that $\Delta(T) \leq \Delta(G)$. □

For the case in which $r > 3$, we find a L_r^{k+1} minor by first determining a bound on the number of pendant vertices in any resolved tree (Lemma 6) and then showing that L_r^{k+1} is a minor of any resolved tree with a sufficiently large number of pendant vertices (Lemma 7).

Lemma 6 *For any resolved tree T, if $|V(T)| \geq 2$, then $|\text{pend}(T)| \geq \frac{1}{2}|V(T)| + 1$.*

Proof. The proof proceeds by induction on the number of pendant vertices of T. When $|\text{pend}(T)| = 2$, $|\text{pend}(T)| = |V(T)| = 2$, and the base case follows. We now suppose that the lemma holds for any tree with fewer than n pendant vertices and show that it holds for a tree T with n pendant vertices, $n > 2$.

Suppose v is a pendant vertex in T, and w is the neighbor of v in T. If w has degree greater than 3, we form T' from T by deleting v. Clearly $|\text{pend}(T')| = |\text{pend}(T)| - 1$ and $|V(T')| = |V(T)| - 1$; applying the induction hypothesis to T' yields $|\text{pend}(T')| \geq \frac{1}{2}|V(T')| + 1$, from which the result follows by simple algebra. If w has degree equal to 3, we let T' be the graph formed by deleting v and contracting one of the edges incident on w. In this case $|\text{pend}(T')| = |\text{pend}(T)| - 1$, $|V(T')| = |V(T)| - 2$, and applying the induction hypothesis to T' yields $|\text{pend}(T')| \geq \frac{1}{2}|V(T')| + 1$, and, as before, the result follows. □

Lemma 7 *Let k and r be integers, $k \geq 0$ and $r \geq 3$. Any resolved tree T where $\Delta(T) \geq r$, $m = \max\{r-1, \Delta(T)\}$ and $|\text{pend}(T)| > m(1+m)(k(2m-2)^2+1)$ contains L_r^{k+1} as a minor. Thus, if L_r^{k+1} is not a minor of a resolved tree T, then T will have at most $O(km^4)$ leaves.*

Proof. We construct a tree U from T by iteratively finding and contracting an edge (u, v), $u, v \notin \text{pend}(T)$, where $\deg_T(u) + \deg_T(v) \leq r + 1$. Any vertex in U that is not the result of a contraction has degree at most $\Delta(T)$. A vertex in U that is the result of a contraction of an edge e will have degree at most $r - 1$, as the sum of the degrees of the endpoints of e is at most $r + 1$. We conclude that $\Delta(U) \leq \max\{r-1, \Delta(T)\} = m$. Clearly $\text{pend}(U) = \text{pend}(T)$ and for any edge $(u, v) \in E(U)$, $u, v \notin \text{pend}(U)$, it must be the case that $\deg_U(u) + \deg_U(v) \geq r + 2$. Since $U \preceq T$, to complete the proof of the lemma it will suffice to prove that $L_r^{k+1} \preceq U$.

We let $Q = V(U) - \text{pend}(U)$. By Lemma 3, $|Q| \geq \frac{|\text{pend}(U)|}{\Delta(U)} > (1+m)(k(2m-2)^2+1)$. As a consequence of Lemma 1, $U[Q]$ contains a matching M where $|M| \geq \frac{|Q|}{\Delta(U)+1} > k(2m-2)^2 + 1$.

We form a new tree U' from U by contracting all the edges in M. Since the matching consists of edges in $U[Q]$, for each edge $(u, v) \in M$, neither u nor v is in $\text{pend}(U)$. By the properties of U shown in the first paragraph of this proof, we can conclude that for any edge $(u, v) \in M$, $\deg_U(u) + \deg_U(v) \geq r + 2$. Consequently, the contraction of an edge in M will result in a new vertex of degree at least r in U'. For R the set of vertices in U' with degree at least r, clearly $|R| \geq |M|$, or $|R| > k(2m-2)^2 + 1$.

Since the degree of a vertex in U' formed by contracting an edge (u, v) will be $\deg_U(u) + \deg_U(v) - 2$, and $\Delta(U) \leq m$, we can conclude that its degree is at most $2m - 2$. If a vertex is not an endpoint of a contracted edge, its degree in U' will be the same as its degree in U and therefore it will be at most $\Delta(T)$. As $\Delta(T) \geq 3$, we can conclude that $\Delta(U') \leq 2m - 2$, or $|R| > k(\Delta(U')^2 + 1)$. Applying Lemma 2 to U', we conclude $L_r^{k+1} \preceq U'$, and since $U' \preceq U$, the proof is complete. $\square$

Proof of Theorem 1. For each value of r, we prove the theorem by contradiction, assuming that $|V(G)| > f(k, r, D, C)$ and showing that as a consequence $L_r^{k+1} \preceq G$.

When $r = 1$, the proof is a direct consequence of Lemma 1.

When $r = 2$, in order to apply Lemma 2 for $r = 2$, we need to find a tree T which is a minor of G and such that T has a large number of vertices of degree at least two. We assume that $|V(G)| > f(k, r, D, C)$. For T any spanning tree of G, clearly $\Delta(T) \leq \Delta(G)$. By Lemma 3, we observe that $|V(T) - \text{pend}(T)| \geq \frac{|V(T)|}{\Delta(T)+1} = \frac{|V(G)|}{\Delta(T)+1} \geq \frac{|V(G)|}{\Delta(G)+1} \geq \frac{|V(G)|}{D+1} \geq k(D^2 + 1) + 1 \geq k(\Delta(G)^2 + 1) + 1 \geq k(\Delta(T)^2 + 1) + 1$. Since each internal vertex in T has degree at least two, $|V(T) - \text{pend}(T)|$ is the set of vertices of degree at least two, and hence by Lemma 2 for $r = 2$, $L_r^{k+1} \preceq T \preceq G$, as needed.

For the case in which $r = 3$, we assume that $|V(G)| > f(k, r, D, C)$ and apply Lemmas 4 and 5 to conclude that G contains as a minor a resolved tree T such that $\Delta(T) \leq \Delta(G)$ and $|V(T)| > \frac{2|V(G)|}{5(\text{chain}(G)\Delta(G)+2)} \geq \frac{2|V(G)|}{5(CD+2)} \geq (D+1)(k(D^2+1)+1))$. Each internal vertex of T has degree at least 3. By Lemma 3, $|V(T) - \text{pend}(T)| \geq \frac{|V(T)|}{\Delta(T)+1} \geq \frac{|V(T)|}{\Delta(G)+1} \geq \frac{|V(T)|}{D+1} \geq (k(D^2+1)+1) \geq (k(\Delta(G)^2+1)+1) \geq (k(\Delta(T)^2+1)+1)$. By Lemma 2, this time for $r = 3$, we conclude that $L_r^{k+1} \preceq T \preceq G$, as needed to obtain a contradiction.

Finally, when $r > 3$ we apply Lemmas 4, 5, and 6 to establish the fact that G contains as a minor a resolved tree T where $|\text{pend}(T)| > \frac{|V(G)|}{5(\text{chain}(G)\Delta(G)+2)} + 1 \geq \frac{|V(G)|}{5(CD+2)} + 1 \geq \gamma(1+\gamma)(k((2\gamma - 2)^2 + 1) + 1)$. By applying Lemma 7, we now obtain $L_r^{k+1} \preceq T \preceq G$. $\square$

We remark that an easy corollary of Lemmas 5 and 6 is that any connected resolved graph of at least $10k$ vertices contains a spanning tree with at least k leaves. This remark can improve the time complexity for the algorithm solving the k-Leaf Spanning Tree problem ([DF95], [DF99, pages 40–42]), from $O(n + (2k)^{4k})$ to $O(n + (10k)^{2k})$.

4 Sizes of obstructions

Our goal is to prove Theorem 2, which gives a bound on the size of a graph in $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ when $\mathcal{G}$ is a bounded degree minor-closed disjoint-union-closed graph class. To this end, we prove a series of useful lemmas.

Lemma 8 *If k is a nonnegative integer and $\mathcal{G}$ is a bounded degree minor-closed graph class, then $L^{k+1}_{\Delta(\mathcal{G})+1}$ cannot be the proper minor of any graph in $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$.*

Proof. Suppose instead that $L^{k+1}_{\Delta(\mathcal{G})+1}$ is the proper minor of a graph in $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$. Since $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ is defined to be a minor-minimal set of obstructions for $\mathcal{W}_k(\mathcal{G})$, $L^{k+1}_{\Delta(\mathcal{G})+1} \in \mathcal{W}_k(\mathcal{G})$. By the definition of $\mathcal{W}_k(\mathcal{G})$, $L^{k+1}_{\Delta(\mathcal{G})+1}$ must contain a set S, $|S| \leq k$, such that $G' = L^{k+1}_{\Delta(\mathcal{G})+1}[V(L^{k+1}_{\Delta(\mathcal{G})+1}) - S] \in \mathcal{G}$. Clearly, $\Delta(G') \leq \Delta(\mathcal{G})$ and therefore $K_{1,\Delta(\mathcal{G})+1}$ cannot be a subgraph of G'. S must then contain at least one vertex in each of the disjoint copies of $K_{1,\Delta(\mathcal{G})+1}$ as a subgraph in $L^{k+1}_{\Delta(\mathcal{G})+1}$, as otherwise G' contains a copy of $K_{1,\Delta(\mathcal{G})+1}$ as a subgraph. Since $|S| \leq k$ and the number of copies of $K_{1,\Delta(\mathcal{G})+1}$ is $k+1$, we obtain a contradiction. $\square$

Lemma 9 *If k is a nonnegative integer and $\mathcal{G}$ is a bounded degree minor-closed graph class, then $\Delta(\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))) \leq k + \Delta(\mathcal{G}) + 1$.*

Proof. We consider a graph $H \in \mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ and an edge $e \in E(H)$. We form a new graph $H' = (V(H), E(H) - \{e\})$ by removing e from H. Since H' is smaller than H, $H \in \mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$, and $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ is a set of minor-minimal elements, we can conclude that $H' \in \mathcal{W}_k(\mathcal{G})$. Consequently, by the definition of $\mathcal{W}_k(\mathcal{G})$, we can partition $V(H')$ into sets S_1 and S_2 where $|S_1| \leq k$ and $H'[S_2] \in \mathcal{G}$. It is not difficult to see that $\Delta(H') \leq \Delta(H'[S_2]) + |S_1|$. The result now follows from the facts that $\Delta(H'[S_2]) \leq \Delta(\mathcal{G})$, $|S_1| \leq k$, and $\Delta(H) \leq \Delta(H') + 1$. $\square$

The idea of the following lemma is taken from work by Dinneen [Din97].

Lemma 10 *For any minor-closed disjoint-union-closed graph class $\mathcal{G}$, if a graph G in $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ is the disjoint union of two nonempty graphs G_1 and G_2, then there exist two positive integers k_1 and k_2 such that $k_1 + k_2 = k + 1$ and $G_i \in \mathrm{ob}(\mathcal{W}_{k_i - 1}(\mathcal{G})), i = 1, 2$.*

Proof. For $i = 1, 2$, let k_i be the minimum l for which $G_i \in \mathrm{ob}(\mathcal{W}_l(\mathcal{G}))$ and notice that, by Lemma 15, $k_i \geq 1$. We let S_i be a subset of $V(G_i)$ of size k_i whose removal from G_i results in a graph in $\mathcal{G}$. The removal from G of $S_1 \cup S_2$ thus results in the disjoint union of two graphs in $\mathcal{G}$, which by closure yields a graph in $\mathcal{G}$. Therefore, $G \in \mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ implies that $k + 1 \leq |S_1 \cup S_2| = |S_1| + |S_2| = k_1 + k_2$. Since $G \in \mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$, clearly $G \in \mathcal{W}_{k+1}(\mathcal{G})$, and hence there exists a set $S \subseteq V(G)$ of size $k + 1$ whose removal from G results in a graph in $\mathcal{G}$. For

$V_i = S \cap V(G_i), i = 1, 2$, $|V_i| \geq k_i$, as the removal of V_i from G_i results in a graph that is in $\mathcal{G}$. Therefore $k + 1 = |S| = |V_1| + |V_2| \geq k_1 + k_2$. We conclude that $k_1 + k_2 = k + 1$.

To prove that $G_i \in \mathrm{ob}(\mathcal{W}_{k_i}(\mathcal{G})), i = 1, 2$, we suppose the contrary; without loss of generality, we assume that $H \in \mathrm{ob}(\mathcal{W}_{k_1}(\mathcal{G}))$ is a proper minor of G_1. Clearly, $G' = H \cup G_2$ is a proper minor of G and therefore there exists a set $S' \subseteq V(H) \cup V(G_2), |S'| \leq k$ whose removal from G' results in a graph in $\mathcal{G}$. As $H \in \mathrm{ob}(\mathcal{W}_{k_1}(\mathcal{G}))$, at least k_1 of the vertices of S' are in H. Therefore, $V' = S' \cap V(G_2)$ contains at most $k - k_1$ vertices and $G_2[V(G_2) - V'] \in \mathcal{G}$. As discussed in the previous paragraph, $G_1[V(G_1) - S_1] \in \mathcal{G}$, and hence by removing $S_1 \cup V'$ from G we produce the disjoint union of two graphs in $\mathcal{G}$, a member of $\mathcal{G}$. Since $|S_1 \cup V'| \leq k_1 + (k - k_1) = k$, we have obtained a contradiction, as $G \in \mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$. □

The lemma below is now easily proved by induction.

Lemma 11 *For any minor-closed disjoint-union-closed graph class $\mathcal{G}$, if a graph in* $\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ *is the disjoint union of r nonempty connected graphs $C_1, \ldots, C_r$, then there exists an integer $r \leq k$ and a partition of $k+1$ into r integers $k_1, \ldots, k_r$ such that for $i = 1, \ldots, r$, $C_i \in \mathrm{ob}(\mathcal{W}_{k_i-1}(\mathcal{G}))$.*

The following two lemmas are useful not only in the proof of Theorem 2, but in the proof of correctness of the algorithm of Section 5.

Lemma 12 *For any minor-closed graph class $\mathcal{G}$, for any $a \geq \mathrm{chain}(\mathrm{ob}(\mathcal{G}))$, and for any $G \in \mathcal{G}$, the subdivision of any edge in any a-chain of G results in a graph that is also a member of $\mathcal{G}$.*

Proof. Suppose instead that there exists a class $\mathcal{G}$, an $a \geq \mathrm{chain}(\mathrm{ob}(\mathcal{G}))$, a graph $G \in \mathcal{G}$, and an edge e in an a-chain of G such that for H the graph obtained from G by subdividing e, H is not in $\mathcal{G}$.

Since H is not in $\mathcal{G}$, by definition there exists a graph $H' \in \mathrm{ob}(\mathcal{G})$ such that H' is a proper minor of H (as $\mathrm{chain}(H) \geq a+1$ and $a \geq \mathrm{chain}(\mathrm{ob}(\mathcal{G}))$, clearly H is not a member of $\mathrm{ob}(\mathcal{G})$). We now consider the operations performed on H to obtain H', and in particular the set of vertices A in H appearing in the $(a+1)$-chain formed by subdividing e. If any of the vertices or edges in A were to be deleted or contracted, the result would be a minor of G, and hence a member of $\mathcal{G}$. Consequently, the $(a+1)$-chain A appears in both H and H', resulting in the fact that $\mathrm{chain}(H') \geq a+1$, contradicting the inequality $a \geq \mathrm{chain}(\mathrm{ob}(\mathcal{G}))$. □

Lemma 13 *For any minor-closed graph class $\mathcal{G}$, and for any $k \geq 0$,* $\mathrm{chain}(\mathrm{ob}(\mathcal{W}_k(\mathcal{G}))) \leq (k+1)\mathrm{chain}(\mathrm{ob}(\mathcal{G}))$.

Proof. Suppose instead that a graph $H \in \mathrm{ob}(\mathcal{W}_k(\mathcal{G}))$ contains a $(b+1)$-chain where $b \geq (k+1)\mathrm{chain}(\mathrm{ob}(\mathcal{G}))$. For H' the graph formed by contracting any edge e of this chain, H'

contains a b-chain C. Moreover, by the minimality of H, $H' \in \mathcal{W}_k(\mathcal{G})$, and hence there exists a partition of $V(H')$ into sets S_1 and S_2 such that $|S_1| \leq k$ and $H'[S_2] \in \mathcal{G}$.

We now show that by applying Lemma 12, we can use the partition of H' to form a partition of H and obtain a contradiction by showing that $H \in \mathcal{W}_k(\mathcal{G})$. We first observe that there must exist at least chain(ob($\mathcal{G}$)) consecutive vertices of C that are in S_2 (this fact follows from the observations that $b \geq (k+1)$chain(ob($\mathcal{G}$)) and that at most k vertices of the b-chain C are in S_1). In other words, $H'[S_2]$ contains a c-chain where $c \geq$ chain(ob($\mathcal{G}$)). Since e was any edge of the original chain, we can view the resultant vertex as a vertex of our choice in C, namely a vertex in a c-chain in $H'[S_2]$. We can now apply Lemma 12 to $H'[S_2]$ to conclude that $H[V(H) - S_1] \in \mathcal{G}$. Since $|S_1| \leq k$, $H \in \mathcal{W}_k(\mathcal{G})$. $\square$

We are now ready to prove our main theorem bounding the size of obstructions.

Theorem 2 *If $\mathcal{G}$ is a bounded degree minor-closed disjoint-union-closed graph class then* max-size(ob($\mathcal{W}_k(\mathcal{G})$)) $\leq f(k, \Delta(\mathcal{G}) + 1, k + \Delta(\mathcal{G}) + 1, (k+1)\text{chain}(\text{ob}(\mathcal{G})))$.

Proof. It suffices to prove that f bounds the size of connected obstructions in ob($\mathcal{W}_k(\mathcal{G})$). To see this, consider a disconnected obstruction with r components $C_1, \ldots, C_r$. By Lemma 11, there is a partition of $k+1$ into r integers $k_1, \ldots, k_r$ such that for $i = 1, \ldots, r$, $C_i \in \text{ob}(\mathcal{W}_{k_i - 1}(\mathcal{G}))$. If the theorem has been proved for connected obstructions, the size of C_i is bounded by $f(k_i - 1, \Delta(\mathcal{G}) + 1, k_i - 1 + \Delta(\mathcal{G}) + 1, (k_i + 1)\text{chain}(\text{ob}(\mathcal{G})))$. Since f is superadditive in the parameter f and $\sum_{i=1}^{r}(k_i - 1) = k - r + 1 \leq k$, this is bounded above by $f(k, \Delta(\mathcal{G}) + 1, k + \Delta(\mathcal{G}) + 1, (k+1)\text{chain}(\text{ob}(\mathcal{G})))$, as required. (Readers interested in the sharpest possible bounds will note that k can be replaced by $k - r + 1$ in the case of an obstruction with r components.)

For H a connected graph in ob($\mathcal{W}_k(\mathcal{G})$), by Lemma 13, we know that chain(H) $\leq (k+1)$chain(ob($\mathcal{G}$)), and by Lemma 9, we know that $\Delta(H) \leq k + \Delta(\mathcal{G}) + 1$. If $L^{k+1}_{\Delta(\mathcal{G})+1}$ is a minor of H, then since H is connected and $L^{k+1}_{\Delta(\mathcal{G})+1}$ is not, $L^{k+1}_{\Delta(\mathcal{G})+1}$ must be a proper minor of H, contradicting Lemma 8. We can thus conclude that H is $L^{k+1}_{\Delta(\mathcal{G})+1}$-minor free. Therefore, Theorem 1 implies that $|V(H)| \leq f(k, \Delta(\mathcal{G}) + 1, k + \Delta(\mathcal{G}) + 1, (k+1)\text{chain}(\text{ob}(\mathcal{G})))$. $\square$

For $\Delta(\mathcal{G}) = 0$, Theorem 2 implies that the obstructions for the class of graphs with vertex cover at most k have size at most $(k+1)(k+2)$. In the case where $\Delta(\mathcal{G}) = 1$, the upper bound is $(k+3)(k((k+2)^2+1)+1)$ or $O(k^4)$; in the case $\Delta(\mathcal{G}) = 2$, the upper bound is $\frac{5}{2}(k+4)(k((k+3)^2+1)+1)(\text{chain}(\text{ob}(\mathcal{G}))(k+3)+2)$ or $O(\text{chain}(\text{ob}(\mathcal{G}))k^5)$; and when $\Delta(\mathcal{G}) \geq 3$, the upper bound is in $O(\text{chain}(\text{ob}(\mathcal{G}))k^2(k + \Delta(\mathcal{G}))^5)$.

A minor-closed graph class $\mathcal{G}$ has bounded degree if and only if $K_{1,\Delta(\mathcal{G})+1}$ is one of its obstructions. This implies that if t is the size of the biggest obstruction in ob($\mathcal{G}$), $\Delta(G) \leq t$. Since chain(ob($\mathcal{G}$)) $\leq t$, we can now conclude the following.

Corollary 1 *If $\mathcal{G}$ is a bounded degree minor-closed disjoint-union-closed graph class then* $\text{max-size}(\text{ob}(\mathcal{W}_k(\mathcal{G}))) = O(tk^7 + t^6k^2)$ *where* $t = \text{max-size}(\text{ob}(\mathcal{G}))$.

The following two lemmas are useful in the proof of correctness of the algorithm in Section 5.

Lemma 14 *All obstructions of a minor-closed graph class $\mathcal{G}$ are connected graphs if and only if $\mathcal{G}$ is closed under disjoint union.*

Proof. We prove the contrapositive of each direction. Suppose instead that $\mathcal{G}$ is not closed under disjoint union; we let G_1 and G_2 be members of $\mathcal{G}$ such that $G = G_1 \cup G_2 \notin \mathcal{G}$. Since $G \notin \mathcal{G}$, G must contain an obstruction H of $\mathcal{G}$ as a minor. If H is connected, then it must be a minor of either G_1 or G_2, contradicting the fact that G_1 and G_2 are in $\mathcal{G}$. We can then conclude that H must be disconnected.

We now suppose that there exists a disconnected obstruction H of $\mathcal{G}$ with disjoint components H_1 and H_2. Since H is an obstruction, any minor of H is in $\mathcal{G}$, and hence both H_1 and H_2 are in $\mathcal{G}$. We have just demonstrated members of $\mathcal{G}$ whose disjoint union is not in $\mathcal{G}$, as needed to complete the proof. $\square$

The following is an easy corollary of Lemma 14.

Lemma 15 *If $\mathcal{G}$ is a minor-closed disjoint-union-closed graph class, then, for any $k \geq 0$, no graph in $\text{ob}(\mathcal{W}_k(\mathcal{G}))$ contains a member of $\mathcal{G}$ as a connected component.*

Proof. Suppose instead that $G \in \text{ob}(\mathcal{W}_k(\mathcal{G}))$ is the disjoint union of two graphs F and H where $H \in \mathcal{G}$. Since G is minor-minimal, $F \in \mathcal{W}_k(\mathcal{G})$, and hence there is a set S of at most k vertices such that $F[V(F) - S] \in \mathcal{G}$. The graph $G[V(G) - S]$ is the disjoint union of $F[V(F) - S]$ and H, both of which are members of $\mathcal{G}$. By Lemma 14 this union is also a member of $\mathcal{G}$, contradicting G's membership in $\text{ob}(\mathcal{W}_k(\mathcal{G}))$. $\square$

5 Recognizing graphs within k vertices of a bounded degree minor-closed class

When k is considered to be part of the input and $\mathcal{G}$ is characterized by a nontrivial property, the problem of deciding whether $G \in \mathcal{W}_k(\mathcal{G})$ is NP-complete [LP80]. In contrast, when k is viewed as a parameter and $\mathcal{G}$ is any bounded degree minor-closed disjoint-union-closed graph class, we are able to obtain a fast fixed-parameter tractable algorithm, as shown in this section.

Theorem 3 *Let $\mathcal{G}$ be any minor-closed graph class with the following properties:*

1. *$\mathcal{G}$ contains only graphs of degree bounded by $D \geq 3$;*

2. $\text{chain}(\text{ob}(\mathcal{G})) \leq C$;

3. *all obstructions of $\mathcal{G}$ are connected; and*

4. *checking whether $G \in \mathcal{G}$ can be executed in $O(|V(G)| \cdot \sigma_{\mathcal{G}})$ time where $\sigma_{\mathcal{G}}$ is a constant depending on the class $\mathcal{G}$.*

Then, for any k, there exists an $O((D+k+\sigma_{\mathcal{G}})|V(G)| + \sigma_{\mathcal{G}}(f(k, D+1, D+k, C(k+1)))^k)$-time algorithm that decides if a graph G is in $\mathcal{W}_k(\mathcal{G})$ and, if so, produces a set $S \subseteq V(G), |S| \leq k$, such that $G[V(G) - S] \in \mathcal{G}$.

Proof of Theorem 3. We propose the following algorithm.

1 `Set` A `to the set of vertices in` G `with degree greater than` $D+k$.

2 `If` $|A| > k$ `then return "no" and stop; otherwise set` $H = G[V(G) - A]$.

3 `Set` $k' = k - |A|$.

4 `As long as there exists in` H `a` β`-chain where` $\beta > C(k'+1)$,
`replace it with a` $(C(k'+1))$`-chain.`

5 `Construct graph` J `by removing from` H `all its`
`connected components that are in` $\mathcal{G}$.

6 `If` $|V(J)| \geq f(k', D+1, D+k, C(k'+1))$ `then return "no" and stop.`

7 `Check if` J `contains a solution` S. `If not, return "no" and stop;`
`otherwise, return` $A \cup S$.

To see that the algorithm is correct, we first observe that if there exists a solution, it will contain every vertex of degree greater than $D+k$ and hence $|A| \leq k$ (step 2). The problem is then reduced to the question of whether $H = G[V(G) - A]$ is in $\mathcal{W}_{k'}(\mathcal{G})$, for $\Delta(H) \leq D+k$. As a consequence of Lemma 13 and property 2 of $\mathcal{G}$, $\text{chain}(\text{ob}(\mathcal{W}_{k'}(\mathcal{G}))) \leq C(k'+1)$. By combining this fact with Lemma 12 (for $a = C(k'+1)$), since $\mathcal{G}$ is minor-closed we can conclude that step 4 preserves membership in $\mathcal{W}_{k'}(\mathcal{G})$ and results in a graph H where $\text{chain}(H) \leq C(k'+1)$. Property 3 of $\mathcal{G}$ and Lemma 14 imply that membership in $\mathcal{W}_{k'}(\mathcal{G})$ is invariant under the removal from H of all its connected components that are members of $\mathcal{G}$. In this way, the problem is reduced to determining whether $J \in \mathcal{W}_{k'}(\mathcal{G})$ where $\text{chain}(J) \leq C(k'+1)$ and $\Delta(J) \leq D+k$, where for $J_1, \ldots, J_m$ the connected components of J, $J_i \notin \mathcal{G}$. Each J_i must then contain at least one vertex of a possible solution S.

If $J \in \mathcal{W}_{k'}(\mathcal{G})$, there exists a partition $(S_1, \ldots, S_m)$ of S where for $i = 1 \ldots m$, $S_i = S \cap V(J_i)$, $k_i = |S_i|$, $J_i[V(J_i) - S_i] \in \mathcal{G}$, and therefore $J_i \in \mathcal{W}_{k_i}(\mathcal{G})$. By Lemma 8, for $i = 1, \ldots, m$, J_i is $L_{D+1}^{k_i+1}$-minor free. Since for $i = 1, \ldots, m$, $\Delta(J_i) \leq D+k$ and $\text{chain}(J_i) \leq C(k'+1)$, we apply Theorem 1 to show that $V(J_i) \leq f(k_i, D+1, D+k, C(k'+1))$. Since J is composed of its connected components, $|V(J)| \leq \sum_{i=1,\ldots,m} f(k_i, D+1, D+k, C(k'+1))$. The maximum value of $|V(J)|$ is achieved when $m=1$, due to the restrictions that $\forall_{1\leq i\leq m}\ k_i \geq 1$ and $\sum_{i=1}^{m} k_i = k'$, as f is a monotonically increasing function and hence $f(k_1)+f(k_2) \leq f(k_1+k_2)$. We can then conclude that $|V(J)| \leq f(k', D+1, D+k, C(k'+1))$, justifying step 6.

To determine the complexity of the algorithm, we first observe that steps 1 through 3 run in $O(|V(G)|(D+k))$ time and step 4 in $O(|V(G)|)$ time. As a consequence of property 4 of $\mathcal{G}$, step 5 can be executed in $O(|V(G)|\sigma_{\mathcal{G}})$ time. Step 7 can be implemented by checking whether $J[V(J)-S'] \in \mathcal{G}$ for all sets $S' \subseteq V(J), |S| \leq k'$, in $O(\sigma_{\mathcal{G}}(f(k', D+1, D+k, C(k'+1)))^{k'})$ time. The overall time complexity of the algorithm is thus in $O((D+k+\sigma_{\mathcal{G}})|V(G)| + \sigma_{\mathcal{G}}(f(k, D+1, D+k, C(k+1)))^{k})$, as claimed. □

As we mentioned in the end of Section 4, both $\Delta(\mathcal{G})$ and $\text{chain}(\text{ob}(\mathcal{G}))$ are bounded by $t = \text{max-size}(\text{ob}(\mathcal{G}))$ and hence the complexity of Theorem 3 can be rewritten as $O((t+k+\sigma_{\mathcal{G}})|V(G)| + \sigma_{\mathcal{G}}(f(k, t+1, t+k, t(k+1)))^{k}) = O((t+k+\sigma_{\mathcal{G}})|V(G)| + \sigma_{\mathcal{G}}(O(tk^2(t+k)^5)^k)$.

Suppose now that $\mathcal{G}$ satisfies conditions 1–3 of Theorem 3 and that $\text{ob}(\mathcal{G})$ is known. As $K_{1,\Delta(\mathcal{G})+1} \in \text{ob}(\mathcal{G})$, all the graphs in $\mathcal{G}$ have pathwidth bounded by D [BRST91]. Using standard techniques on graphs with bounded treewidth [Arn85], one can construct an algorithm that checks whether $H \leq G$ in $O(\chi(t, |V(H)|)|V(G)|)$ time, where t is a bound on the treewidth of G and χ is a super-polynomial function. Consequently, if we know $\text{ob}(\mathcal{G})$, then each of the checks for membership in $\mathcal{G}$ applied in steps 5 and 7 can be done in $O(|\text{ob}(\mathcal{G})| \cdot \chi(D, \text{max-size}(\text{ob}(\mathcal{G})))|\mathcal{G}|)$ time and $\sigma_{\mathcal{G}}$ can be replaced by $|\text{ob}(\mathcal{G})| \cdot \chi(t, \text{max-size}(\text{ob}(\mathcal{G})))$. As a result, condition 4 will be satisfied.

6 Recognizing graphs within k vertices of a bounded degree class

In this section we extend our techniques to provide a solution to a generalization of the vertex cover problem that is not closed under taking of minors. Let $\mathcal{G}_D$ denote the class of graphs with maximum degree at most D. We define the problem ALMOST D-BOUNDED GRAPH as deciding whether or not a graph G is in $\mathcal{W}_k(\mathcal{G}_D)$, and if so, finding a set $S \subseteq V(G)$ where $|S| \leq k$ and $G[V(G)-S] \in \mathcal{G}_D$. If $D=0$, the problem is simply VERTEX COVER. To the best of our knowledge, the best algorithm so far for the general problem can be derived as a subcase of the $\Pi_{i,j,k}$ GRAPH MODIFICATION PROBLEM [Cai96]. The running time obtained is $O((D+1)^k|V(G)|^{D+1})$. The function given below improves on this bound (Theorem 4) but the algorithm which uses it as a subroutine is even faster.

Function $\mathsf{Search}_D(G,k)$

1 If $k<0$ then return "no".

2 If $\Delta(G) > D+k$, then let v be any vertex of degree greater than $D+k$
  if $\mathsf{Search}_D(G-v,k-1) =$"no", then return "no"
  otherwise, return $\{v\} \cup \mathsf{Search}_D(G-v,k-1)$.

3 If $\Delta(G) \leq D$, then return $\emptyset$.

4 Set v to be any vertex such that $D < d_G(v) \leq D+k$.
  (If there exists no such vertex in G then return "no".)

5 If for all $u \in N_G(v) \cup \{v\}$ $\mathsf{Search}_D(G-u,k-1) =$"no",
  then return "no"
  otherwise for any $u \in N_G(v) \cup \{v\}$, return $\mathsf{Search}_D(G-u,k-1)$.

Algorithm

1 Set A to the set of vertices in G with degree greater than $D+k$.

2 If $|A| > k$ then return "no" and stop; otherwise set $H = G[V(G)-A]$.

3 Set $k' = k - |A|$.

4 Set F to the vertices in $V(H)$ that have degree at most D
  and set $C = V(H) - F$.

5 Set B to the vertices of F that are adjacent, in H, to vertices of C.

6 If $|B \cup C| \geq (D+k+1)k' + (D+k)(D+k+1)k'$ then return "no" and stop.

7 If $\mathsf{Search}_D(H[B\cup C],k') =$"no" then return "no" and stop;
  otherwise, return $A \cup \mathsf{Search}_D(H[B \cup C],k')$ and stop.

Theorem 4 *For $\mathcal{G}_D$ the class of graphs with maximum degree at most $D \geq 0$, function $\mathsf{Search}_D(G,k)$ solves* Almost D-bounded graph *in time $O(|V(G)|(D+k)^{k+1})$.*

Proof. The correctness of $\mathsf{Search}_D(G,k)$ can be shown by justifying each line. If a solution exists, it must contain each vertex v that has degree at least $D+k$, and hence $G \in \mathcal{W}_k(\mathcal{G}_D)$ if and only if $G-v \in \mathcal{W}_{k-1}(\mathcal{G}_D)$ (step 2). Any solution must contain either each vertex v of degree at least D or a vertex in each such v's neighborhood, so that $G \in \mathcal{W}_k(\mathcal{G}_D)$ if and only if for some $u \in N_G(v) \cup \{v\}$ $G-u \in \mathcal{W}_{k-1}(\mathcal{G}_D)$ (step 5).

Each time $\mathsf{Search}_D(G,k)$ is recursively called it requires $O(|V(G)|(D+k))$ time to make the degree checks of steps 2 and 4. As a consequence of step 5, each node of the search tree followed by the recursion has at most $D+k+1$ children. As the depth of this tree cannot be more than k, the number of recursive calls is at most $O((D+k+1)^k)$, yielding an overall running time of $O(|V(G)|(D+k)^{k+1})$. $\square$

The following corollary of Theorem 4 is a generalization of the solution to the Log Vertex Cover problem (the case $D = 0$ [PY96] [DF99, page 30]).

Corollary 2 *There is a polynomial-time algorithm that decides, given a graph G and an integer $D \geq 0$ whether G contains a set of size $O(\log |V(G)|)$ whose removal results in a graph of degree bounded by D.*

In order to prove the correctness of the algorithm, we need the following lemma.

Lemma 16 *Let H' be a graph containing a set S of size at most k such that $\Delta(H'[V(H') - S]) \leq D$, $\Delta(H') \leq D + k$, and any vertex of degree at most D is adjacent to a vertex of degree greater than D. Then $V(H') \leq (D + k + 1)k' + (D + k)(D + k + 1)k'$.*

Proof. We let B' be the vertices in $V(H') - S$ of degree at most D and let $C' = V(H') - S - B'$. Each vertex in C' will be adjacent to at least one vertex in S as otherwise $H'[V(H') - S]$ will contain a vertex of degree greater than D. Since $\Delta(S) \leq \Delta(H') \leq k + D$, C' will contain at most $D + k$ neighbours of each vertex in S, or a total of at most $k'(D + k)$ vertices. We have established that $|C'| + |S| \leq (D + k + 1)k'$. We let J be the vertices of B' that are adjacent to vertices in $C' \cup S$ and observe that since each vertex in $C' \cup S$ can have at most $k + D$ neighbours in J, $|J| \leq (D + k)(|C'| + |S|) \leq (D + k)(D + k + 1)k'$. The assumption in the statement of the lemma requires that any vertex in B' be adjacent to a vertex in C', and hence any vertex in B' is adjacent to a vertex in $C' \cup S$. Therefore $B' = J$ and thus $V(H') = |C'| + |S| + |B'| \leq (D + k + 1)k' + (D + k)(D + k + 1)k'$. $\square$

Theorem 5 *For $\mathcal{G}_D$ the class of graphs with maximum degree at most $D \geq 0$, the algorithm above solves* ALMOST D-BOUNDED GRAPH *in time $O(|V(G)|(D + k) + k(D + k)^{k+3})$.*

Proof. If $G \in \mathcal{W}_k(\mathcal{G}_D)$, then there exists a set $S, |S| \leq k$ where $\Delta(G[V(G) - S]) \leq D$, and hence all the vertices of G that are not in S have degree at most $D + k$, yielding $A \subseteq S$. If the set A has more than k vertices, then G cannot be a member of $\mathcal{W}_k(\mathcal{G}_D)$ (steps 1 and 2). It is now easy to see that $G \in \mathcal{W}_k(\mathcal{G}_D)$ if and only if $H \in \mathcal{W}_{k-|A|}(\mathcal{G}_D)$. Therefore, the problem is reduced to the question of whether or not $H \in \mathcal{W}_{k'}(\mathcal{G}_D)$, where $\Delta(H) \leq D + k$. The vertices of $V(H)$ that are not in $B \cup C$ have all degree at most D and are adjacent to vertices that have degree at most D. As a consequence $H \in \mathcal{W}_{k'}(\mathcal{G}_D)$ if and only if $H[B \cup C] \in \mathcal{W}_{k'}(\mathcal{G}_D)$. The graph $H' = H[B \cup C]$ has maximum degree at most $D + k$ and any vertex of degree at most D is adjacent to a vertex of degree greater than D in H'. By Lemma 16, if S is a solution for H' then $|V(H')| = |B \cup C| \leq (D + k + 1)k' + (D + k)(D + k + 1)k'$ (step 6). By Theorem 4, step 7 requires time $O(|V(H')|(D + k)^{k+1}) = O(k(D + k)^{k+3})$. Since steps 1–6 can be trivially implemented in $O((D + k)|V(G)|)$ time, the total running time is as claimed. $\square$

7 Future work

Ideally, we would like to remove the restriction of closure under disjoint union from our results. As a consequence of Theorem 11, for $\mathcal{G}$ a minor-closed disjoint-union-closed graph class, an obstruction of $\mathcal{W}_k(\mathcal{G})$ can have at most $k+1$ components. Unfortunately, if $\mathcal{G}$ is not restricted to be a disjoint-union-closed graph class, $\mathcal{G}$ may have disconnected obstructions, ruining the proof of Lemma 15. It would be an interesting result, requiring new techniques, to determine an upper bound on the number of connected components in an obstruction of $\mathcal{W}_k(\mathcal{G})$ when $\mathcal{G}$ is not a disjoint-union-closed graph class. It would also be nice to extend the approach of Section 6 to find a better algorithm for the general $\Pi_{i,j,k}$-Graph Modification Problem.

References

[Arn85] Stefan Arnborg. Efficient algorithms for combinatorial problems on graphs with bounded decomposability – A survey. *BIT*, 25:2–23, 1985.

[BFR98] R. Balasubramanian, Michael Fellows, and Venkatesh Raman. An improved fixed-parameter algorithm for vertex cover. *Inform. Proc. Letters*, 65:163–168, 1998.

[BG93] Jonathan F. Buss and Judy Goldsmith. Nondeterminism within P. *SIAM J. Computing*, 22:560–572, 1993.

[BRST91] Daniel Bienstock, Neil Robertson, Paul D. Seymour, and Robin Thomas. Quickly excluding a forest. *J. Comb. Theory Series B*, 52:274–283, 1991.

[Cai96] Leizhen Cai. Fixed-parameter tractability of graph modification problems for hereditary properties. *Inform. Proc. Letters*, 58:171–176, 1996.

[CCDF97] Liming Cai, Jianer Chen, Rodney Downey, and Michael Fellows. Advice classes of parameterized tractability. *Annals of Pure and Applied Logic*, 84:119–138, 1997.

[CKJ99] J. Chen, I.A. Kanj, and W. Jia. Vertex cover: Further observations and further improvements. In *Proceedings of the 25th International Workshop on Graph-Theoretical Concepts in Computer Science*, pages 313–324. Springer-Verlag, 1999.

[DF95] Rodney G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness I: Basic results. *SIAM J. Comput.*, 24:873–921, 1995.

[DF99] Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.

[DFS99] Rodney G. Downey, Michael R. Fellows, and Ulrike Stege. Computational tractability: the view from Mars. *Bulletin of the European Association of Theoretical Computer Science*, 69:73–97, 1999.

[Din95] Michael J. Dinneen. *Bounded Combinatorial Width and Forbidden Substructures.* PhD thesis, Department of Computer Science, University of Victoria, 1995.

[Din97] Michael J. Dinneen. Too many minor order obstructions (for parametrized lower ideals). *Journal of Universal Computer Science*, 3(11):1199–1206, 1997.

[FL88] Michael R. Fellows and Michael A. Langston. Nonconstructive tools for proving polynomial-time decidability. *J. ACM*, 35:727–739, 1988.

[FL94] Michael R. Fellows and Michael A. Langston. On search, decision, and the efficiency of polynomial-time algorithms. *J. Comp. Syst. Sc.*, 49(3):769–779, 1994.

[FPT95] Martin Farach, Teresa Przytycka, and Mikkel Thorup. On the agreement of many trees. *Inform. Proc. Letters*, 55:297–301, 1995.

[FRS87] Harvey Friedman, Neil Robertson, and Paul D. Seymour. The metamathematics of the graph minor theorem. *Contemporary Mathematics*, 65:229–261, 1987.

[KST99] Haim Kaplan, Ron Shamir, and Robert E. Tarjan. Tractability of parameterized completion problems on chordal, strongly chordal, and proper interval graphs. *SIAM J. Comput.*, 28(5):1906–1922, 1999.

[Lag93] Jens Lagergren. An upper bound on the size of an obstruction. In Neil Robertson and Paul Seymour, editors, *Graph Structure Theory, Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference, Seattle WA, June 1991*, pages 601–621, Providence, RI, 1993. American Math. Soc. Contemp. Math. 147.

[Lag98] Jens Lagergren. Upper bounds on the size of obstructions and intertwines. *J. Combin. Theory Ser. B*, 73:7–40, 1998.

[Lov79] Laszlo Lovasz. *Combinatorial Problems and Exercises.* North Holland, 1979.

[LP80] J. M. Lewis and Christos H. Papadimitriou. The node-deletion problem for hereditary properties is NP-complete. *J. Comp. Syst. Sc.*, 20:219–230, 1980.

[LP98] Michael A Langston and Barbara C. Plaut. On algorithmic applications of the immersion order. An overview of ongoing work presented at the Third Slovenian International Conference on Graph Theory. *Discrete Mathematics*, 182(1-3):191–196, 1998.

[Meh84] Kurt Mehlhorn. *Graph Algorithms and NP-Completeness*, volume 2 of *Data Structures and Algorithms*. Springer Verlag, Berlin, 1984.

[MR99] Meena Mahajan and Venkatesh Raman. Parameterizing above guaranteed values: maxsat and maxcut. *J. Algorithms*, 31(2):335–354, May 1999.

[Nie98] Rolf Niedermeier. Some prospects for efficient fixed parameter algorithms. In *Proceedings of the 25th Conference on Current Trends in Theory and Practice of Informatics (SOFSEM'98)*, volume 1521 of *Lecture Notes in Computer Science*, pages 168–185, 1998.

[NR99] Rolf Niedermeier and Peter Rossmanith. Upper bounds for vertex cover further improved. In *C. Meinel, S. Tison, editors, Proceedings of the 16th Symposium on Theoretical Aspects of Computer Science (STACS'99)*, volume 1563 of *Lecture Notes in Computer Science*, pages 561–570, 1999.

[PY96] Christos H. Papadimitriou and Mihalis Yannakakis. On limited nondeterminism and the complexity of the V–C dimension. *J. Comp. Syst. Sc.*, 53:161–170, 1996.

[Ram95] Siddharthan Ramachandramurthi. A lower bound for treewidth and its consequences. In Ernst W. Mayr, Gunther Schmidt, and Gottfried Tinhofer, editors, *Proceedings 20th International Workshop on Graph Theoretic Concepts in Computer Science WG'94*, volume 903 of *Lecture Notes in Computer Science*, pages 14–25. Springer Verlag, 1995.

[RS85a] Neil Robertson and Paul D. Seymour. Disjoint paths – A survey. *SIAM J. Alg. Disc. Meth.*, 6:300–305, 1985.

[RS85b] Neil Robertson and Paul D. Seymour. Graph minors — A survey. In I. Anderson, editor, *Surveys in Combinatorics*, pages 153–171. Cambridge Univ. Press, 1985.

[RS95] Neil Robertson and Paul D. Seymour. Graph minors. XIII. The disjoint paths problem. *J. Comb. Theory Series B*, 63:65–110, 1995.

[Thi00] Dimitrios M. Thilikos. Algorithms and obstructions for linear-width and related search parameters. *Discrete Applied Mathematics*, 105:239–271, 2000.

[TUK94] Atsushi Takahashi, Shuichi Ueno, and Yoji Kajitani. Minimal acyclic forbidden minors for the family of graphs with bounded path-width. *Discrete Mathematics*, 127(1/3):293–304, 1994.

[van90] Jan van Leeuwen. Graph algorithms. In *Handbook of Theoretical Computer Science, A: Algorithms and Complexity Theory*, pages 527–631, Amsterdam, 1990. North Holland Publ. Comp.