Excluded Grid Minors and Efficient Polynomial-Time Approximation Schemes

FEDOR V. FOMIN and DANIEL LOKSHTANOV, University of Bergen
SAKET SAURABH, University of Bergen and the Institute of Mathematical Sciences

Two of the most widely used approaches to obtain polynomial-time approximation schemes (PTASs) on planar graphs are the Lipton-Tarjan separator-based approach and Baker's approach. In 2005, Demaine and Hajiaghayi strengthened both approaches using bidimensionality and obtained efficient polynomial-time approximation schemes (EPTASs) for several problems, including Connected Dominating Set and Feedback Vertex Set. In this work, we unify the two strengthened approaches to combine the best of both worlds. We develop a framework allowing the design of EPTAS on classes of graphs with the subquadratic grid minor (SQGM) property. Roughly speaking, a class of graphs has the SQGM property if, for every graph G from the class, the fact that G contains no $t \times t$ grid as a minor guarantees that the treewidth of G is subquadratic in t. For example, the class of planar graphs and, more generally, classes of graphs excluding some fixed graph as a minor, have the SQGM property. At the heart of our framework is a decomposition lemma stating that for "most" bidimensional problems on a graph class $\mathcal{G}$ with the SQGM property, there is a polynomial-time algorithm that, given a graph $G \in \mathcal{G}$ as input and an $\epsilon > 0$, outputs a vertex set X of size $\epsilon \cdot \mathrm{OPT}$ such that the treewidth of $G - X$ is $f(\epsilon)$. Here, OPT is the objective function value of the problem in question and f is a function depending only on ϵ. This allows us to obtain EPTASs on (apex)-minor-free graphs for all problems covered by the previous framework as well as for a wide range of packing problems, partial covering problems and problems that are neither closed under taking minors nor contractions. To the best of our knowledge, for many of these problems—including Cycle Packing, $\mathcal{F}$-Packing, $\mathcal{F}$-Deletion, Max Leaf Spanning Tree, or Partial r-Dominating Set —no EPTASs, even on planar graphs, were previously known.

We also prove novel excluded grid theorems in unit disk and map graphs without large cliques. Using these theorems, we show that these classes of graphs have the SQGM property. Based on the developed framework, we design EPTASs and subexponential time parameterized algorithms for various classes of problems on unit disk and map graphs.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems; G.2.2 [**Graph Theory**]: Graph Algorithms

General Terms: Algorithms, Design, Theory

Additional Key Words and Phrases: Polynomial-time approximation scheme, monadic second-order logic, parameterized complexity, embedded graph, bidimensionality, treewidth, unit disk graph

Preliminary versions of this work appeared in the proceedings of SODA'11 (Fomin et al. 2011a) and SODA'12 (Fomin et al. 2012a).

Authors' addresses: F. V. Fomin and D. Lokshtanov, Department of Informatics, University of Bergen, N-5020 Bergen, Norway; emails: {fomin, daniello}@ii.uib.no; S. Saurabh, Department of Informatics, University of Bergen, N-5020 Bergen, Norway and The Institute of Mathematical Sciences, Chennai - 600113, India; email: saket@imsc.res.in.

https://doi.org/10.1145/3154833

ACM Reference format:
Fedor V. Fomin, Daniel Lokshtanov, and Saket Saurabh. 2018. Excluded Grid Minors and Efficient Polynomial-Time Approximation Schemes. *J. ACM* 65, 2, Article 10 (January 2018), 44 pages.
https://doi.org/10.1145/3154833

1 INTRODUCTION

While many interesting graph problems remain NP-complete even when restricted to planar graphs, the restriction of a problem to planar graphs is usually considerably more tractable algorithmically than the problem on general graphs. Over the last four decades, it has been proved that many graph problems on planar graphs admit subexponential time algorithms (Dorn et al. 2008; Fomin and Thilikos 2006; Lipton and Tarjan 1980), subexponential time parameterized algorithms (Alber et al. 2002; Klein and Marx 2014; Marx 2013), (Efficient) Polynomial Time Approximation Schemes ((E)PTAS) (Baker 1994; Grohe 2003; Dawar et al. 2006; Eisenstat et al. 2012; Eppstein 2000; Gandhi et al. 2004; Khanna and Motwani 1996) and linear kernels (Alber et al. 2004; Bodlaender et al. 2016; Chen et al. 2007). The theory of bidimensionality developed by Demaine and Hajiaghayi (2008a, 2008b) and Demaine et al. (2005b) is able to simultaneously explain the tractability of many planar graphs problems within the paradigms of parameterized algorithms (Demaine et al. 2005a), approximation (Demaine and Hajiaghayi 2005), and kernelization (Fomin et al. 2010). The theory is built on cornerstone theorems from the Graph Minor Theory of Robertson and Seymour, allowing not only the ability to explain the tractability of many problems but also to generalize the results from planar graphs and graphs of bounded genus to graphs excluding a fixed minor. Roughly speaking, a problem is bidimensional if the solution value for the problem on a $k \times k$ grid is $\Omega(k^2)$ and the contraction or removal of an edge does not increase solution value. Many natural problems are bidimensional, including Dominating Set, Feedback Vertex Set, Edge Dominating Set, Vertex Cover, r-Dominating Set, Connected Dominating Set, Cycle Packing, Connected Vertex Cover, and Graph Metric TSP.

A PTAS is an algorithm that takes an instance I of an optimization problem and a parameter $\epsilon > 0$, runs in time $n^{O(f(1/\epsilon))}$, and produces a solution that is within a factor $1 + \epsilon$ of being optimal. A PTAS with runtime $f(1/\epsilon) \cdot n^{O(1)}$ is called an efficient PTAS (EPTAS). Prior to bidimensionality (Demaine and Hajiaghayi 2005), there were two main approaches to design (E)PTASs on planar graphs. The first one was based on the classic Lipton-Tarjan planar separator theorem (Lipton and Tarjan 1979). In the approach of Lipton and Tarjan, we split the input n-vertex graph into two pieces of approximately equal size using a separator of size $O(\sqrt{n})$. Then, we recursively approximate the problem on the two smaller instances and glue the approximate solutions at the separator. This approach was applicable only to problems in which the size of the optimal solutions was at least a constant fraction of n. Recently, the separator-based approach was also used to prove that certain local optimization algorithms are PTASs on minor-free graphs and intersection graphs of various geometric objects (Cabello and Gajser 2015; Chan and Har-Peled 2012; Har-Peled and Quanrud 2015; Mustafa and Ray 2010).

The second, more widely used approach is found in Baker (1994); see also Hochbaum and Maass (1985). The approach is widely used in approximation algorithms and is called the shifting technique, or simply Baker's technique. The main idea is to decompose the planar graph into subgraphs of bounded outerplanarity and then solve the problem optimally in each of these subgraphs using dynamic programming. Later, Eppstein (2000) generalized this approach to work for a larger class of graphs, namely, apex-minor-free graphs. Khanna and Motwani (1996) used Baker's approach in an attempt to syntactically characterize the complexity class of problems admitting PTASs, establishing a family of problems on planar graphs to which it applies. The same kind of approach is also used by Dawar et al. (2006) to obtain EPTASs for every minimization problem definable in first-order logic on every class of graphs excluding a fixed minor. The shifting technique seemed to be limited to "local" graph problems, for which one is interested in finding a vertex/edge set satisfying a property that can be checked by looking at a constant-size neighborhood around each vertex.

Demaine and Hajiaghayi (2005) used bidimensionality theory to strengthen and generalize both approaches. In particular, they strengthened the Lipton-Tarjan approach significantly by showing that for a multitude of problems one can find a separator of size $O(\sqrt{\mathrm{OPT}})$ that splits the optimum solution evenly into two pieces. Here, OPT is the size of an optimum solution. This allowed them to give EPTASs for several problems on planar graphs and, more generally, on apex-minor-free graphs or H-minor-free graphs. Two important problems to which their approach applies are Feedback Vertex Set and Connected Dominating Set. Earlier, only a PTAS and no EPTAS for Feedback Vertex Set on planar graphs was known in Kleinberg and Kumar (2001). In addition, they also generalize Baker's approach by allowing more interaction between the overlapping subgraphs.

Comparing the generalized versions of the two approaches, it seems that each has its strengths and weaknesses. In the generalized Lipton-Tarjan approach of Demaine and Hajiaghayi (2005), one splits the graph into two pieces recursively. To ensure that the repeated application does not "increase" the approximation factor, in each recursive step, one needs to carefully reconstruct the solution from the smaller ones. Additionally, to ensure that the separator splits the optimum solution evenly, the framework of Demaine and Hajiaghayi (2005) requires a constant factor approximation for the problem in question. On the other hand, their generalization of Baker's approach essentially identifies a set X of vertices or edges that interacts in a limited way with the optimum solution, such that the removal of X from the input graph leaves a graph on which the problem can be solved optimally in polynomial time. The set X could be as large as $O(n)$, which makes it difficult to bound the amount of interaction between the set X and the optimum solution in some cases.

In this article, we propose a framework that combines the best of both worlds—the generalized Lipton-Tarjan and generalized Baker's approaches. In particular, we show that for most bidimensional problems, there is a polynomial-time algorithm that, given a graph G and an $\epsilon > 0$, outputs a vertex set X of size $\epsilon \cdot \mathrm{OPT}$ such that the treewidth of $G - X$ is $f(\epsilon)$. Because the *size* of X is bounded, the interaction between X and the optimum solution is bounded trivially. Since X is removed only once, the difficulty faced by a recursive approach vanishes. In our framework to obtain EPTASs, we demand that the problem in question is "reducible," which is nothing else than that the set X can be removed from the graph, disturbing the optimum solution by at most $O(\epsilon \cdot \mathrm{OPT})$. Finally, our algorithm to compute X does not require an approximation algorithm for the problem in question, relying only on a sublinear treewidth bound. For most problems, such a bound can be obtained via bidimensionality, whereas for other problems that are not bidimensional, one can obtain the sublinear treewidth bound directly. Our framework allows one to obtain EPTASs for a broader set of problems, including several packing problems, partial covering problems, and problems that are neither closed under taking minors nor contractions. For many of these problems, no approximation schemes were known prior to our work.

Furthermore, we extend our framework to classes of geometric graphs. (E)PTASs have been studied for various classes of geometric graphs; most of these algorithms use a variation of the shifting techniques. In particular, Hunt et al. (1998) used the shifting technique to give PTASs for a number of problems such as the Independent Set and Dominating Set on unit disk graphs and λ-precision disk graphs. Independently, Erlebach et al. (2005) and Chan (2003) generalized the shifting technique and gave PTASs for the Independent Set and Vertex Cover on disk graphs and on intersection graphs of fat objects. Marx (2008) obtained an EPTAS for Vertex Cover on unit disk graphs. Chen (2001) and Demaine et al. (2005b) used similar approaches to obtain a PTAS for the Independent Set and r-Dominating Set on map graphs. The thesis (van Leeuwen 2009) contains an overview of approximation algorithms on different geometric graphs. As in the case with planar graphs, the limitations of the shifting technique are that it generally applies to only local problems, such as Vertex Cover and variants of Dominating Set, and fails for nonlocal problems such as Feedback Vertex Set and Cycle Packing.

Main results and organization of the article. In Section 2, we collect technical definitions and notations. We start building the framework in Section 3, where we formally define optimization problems that are reducible, bidimensional, separable, and define treewidth-η-modulators. We also define in Section 3 the SQGM and SQGC properties of graph classes. Basically, a graph class $\mathcal{G}$ has the subquadratic grid minor property (SQGM) if, for every graph $G \in \mathcal{G}$, the fact that G does not contain a $t \times t$ grid as a minor yields that the treewidth of G is bounded by a subquadratic function of t. By the theorem of Robertson et al. (1994), planar graphs have the SQGM property. More generally, as it was shown in Demaine and Hajiaghayi (2008c), for every graph H, the class of graphs, excluding H as a minor, has the SQGM property. We are also able to obtain similar results for a more general class of problems on more restricted graph classes. In Section 4, we provide several examples of reducible bidimensional and separable problems.

In Section 5, we prove the technical lemma (Lemma 7) that is at the heart of our main algorithmic result. It allows us to achieve the following "scaling." Let G be a graph from a graph class with the SQGM property. Let X be a vertex set of G such that the treewidth of $G - X$ does not exceed some constant η. Then, in polynomial time, it is possible to scale down X to a set X' of size $\epsilon|X|$ such that the treewidth of $G - X'$ is also bounded by a constant, depending only on ϵ and η. Moreover, every connected component C of graph $G - X'$ has a constant number of neighbors in G. In the terminology of Bodlaender et al. (2016), C is a protrusion.

Lemma 7 allows us to establish algorithmic results for treewidth-η-modulated problems. Basically, for a constant η, an optimization problem Π is treewidth-η-modulated or just η-modulated if, for every graph G, there is a vertex set X such that $|X| = \mathcal{O}(\mathrm{OPT}(\Pi))$ and the treewidth of $G - X$ does not exceed η. In Section 6, we prove a meta-theorem (Theorem 1), which basically says that every η-modulated and sufficiently "well-behaved" graph optimization problem Π has an EPTAS on every hereditary graph class $\mathcal{G}$ with the SQGM property.

In order to apply Theorem 1, we need to establish which optimization problems are η-modulated. Toward this goal, we also prove in Section 6, that for every minor-bidimensional linear-separable problem Π on a graph class with the SQGM property, there exists a constant η such that Π is η-modulated. By our theorem, this immediately implies the existence of EPTAS for many interesting problems, including Vertex Cover, Feedback Vertex Set, Treewidth-η Modulator, and Cycle Packing.

In Section 7, we provide further nontrivial applications of Theorem 1. In particular, we obtain EPTASs for various maximum induced subgraph and packing problems, partial domination and vertex cover problems, where the task is to optimally cover a certain number of vertices or edges.

In Section 8, we show that the SQGM property, and therefore the applicability of Theorem 1, extends beyond minor-closed classes of graphs. We establish two combinatorial theorems showing that unit disk graphs and map graphs that exclude a clique of size t as a subgraph also admit the SQGM property. This immediately transfers all algorithmic results about EPTASs to problems on these classes as well. With some additional work, for several problems including Vertex Cover, Feedback Vertex Set, and Treewidth-η Modulator, we eliminate the K_t-free condition and obtain approximation schemes for unit disk graphs and map graphs. We also prove that Cycle Packing admits a PTAS on unit disk graphs.

As a by-product of the grid exclusion theorems obtained in Section 8, we also obtain subexponential parameterized algorithms on unit disk graphs and map graphs for various problems.

An interesting feature of our algorithms is that they do not require geometric representations of the input graphs. Since recognition of unit disk graphs is NP-hard (Clark et al. 1990) and the best-known exponent of the polynomial bounding the runtime of the map graph recognition algorithm is about 120 (Thorup 1998), the robustness of our algorithms is a serious advantage.

Finally, we explore to which degree our approach can be lifted to other classes of graphs. Our investigations show that it is unlikely that the full power of our approach can be generalized to disk graphs or to unit ball graphs in $\mathbb{R}^d$—intersection graphs of unit-balls in $\mathbb{R}^d$, $d \geq 3$. Specifically, we prove that Feedback Vertex Set on unit-ball graphs in $\mathbb{R}^3$ neither admits a PTAS unless P=NP nor a subexponential time algorithm unless the Exponential Time Hypothesis (ETH) fails.

2 DEFINITIONS AND NOTATIONS

In this section, we give various definitions that we make use of in the article. We start from graph theory definitions.

Let G be a graph with vertex set $V(G)$ and edge set $E(G)$. A graph G' is a *subgraph* of G if $V(G') \subseteq V(G)$ and $E(G') \subseteq E(G)$. For vertex set $X \subseteq V(G)$, the subgraph $G[X]$ is a subgraph of G induced by X or just *induced subgraph* of G if $E(G[X]) = \{uv \in E(G) \mid u, v \in X\}$. For vertex set $S \subseteq V(G)$, we denote by $G - S$ the graph obtained from G by removing the vertices of S, i.e., $G - S = G[V(G) \setminus S]$. For vertex $v \in V(G)$, we also use $G - v$ for $G - \{v\}$. Similarly, for edge set $E \subseteq E(G)$, we use $G - E$ to denote the subgraph of G with edge set $E(G) \setminus E$. For $e \in E(G)$, we use $G - e$ for $G - \{e\}$. For a set $S \subseteq V(G)$, we define $N_G(S)$ to be the *open neighborhood* of S in G, which is the set of vertices from $V(G) \setminus S$ adjacent to vertices of S. The *closed neighborhood* of S is $N_G[S] := N(S) \cup S$. Given a set $S \subseteq V(G)$, we denote by $\partial_G(S)$ the set of all vertices in S that are adjacent in G with vertices not in S. Thus, $N_G(S) = \partial_G(V(G) \setminus S)$. The *distance* $d_G(u, v)$ between two vertices u and v of G is the length of the shortest path in G from u to v. We define $B^r_G(v)$ to be the set of vertices within distance at most r from v, including v itself. For a vertex set S, define $B^r_G(S) = \bigcup_{v \in S} B^r_G(v)$.

A graph class $\mathcal{G}$ is *hereditary* if, for any graph $G \in \mathcal{G}$, all induced subgraphs of G are in $\mathcal{G}$. Throughout this article, K_t denotes a complete graph on t vertices and we say that a graph G is K_t*-free* if G does not contain K_t as an induced subgraph.

Planar, H-minor-free, unit disk and map graphs. In this article, we use the expression *plane graph* for any planar graph drawn in the Euclidean plane $\mathbb{R}^2$ without any edge crossing. We do not distinguish between a vertex of a plane graph and the point of $\mathbb{R}^2$ used in the drawing to represent the vertex or between an edge and the curve representing it. We also consider plane graph G as the union of the points corresponding to its vertices and edges. We call a *face* of G any connected component of $\mathbb{R}^2 \setminus (E(G) \cup V(G))$. The *boundary* of a face is the set of edges incident to it. If the boundary of a face f forms a cycle, then we call it a *cyclic face*. A graph is *planar* if it admits a planar drawing.

Given an edge $e = xy$ of a graph G, the graph G/e is obtained from G by contracting e. That means that the endpoints x and y are replaced by a new vertex $v_{x,y}$, which is adjacent to the old neighbors of x and y (except for x and y). A graph H obtained by a sequence of edge contractions is said to be a *contraction* of G. A graph H is a *minor* of a graph G if H is the contraction of some subgraph of G. Let G, H be two graphs. A subgraph G' of G is said to be a *minor model* of H in G if G' contains H as a minor. We say that a graph G is H*-minor-free* when it does not contain H as a minor. We also say that a graph class $\mathcal{G}$ is H*-minor-free* (or excludes H as a minor) when all its members are H-minor-free. A graph G is an *apex graph* if there exists a vertex v such that $G - v$ is planar. A graph class $\mathcal{G}$ is *apex-minor-free* if there exists an apex graph H

such that $\mathcal{G}$ is H-minor-free. A graph class $\mathcal{G}$ is said to be *minor closed/contraction closed* if every minor/contraction of a graph in $\mathcal{G}$ also belongs to $\mathcal{G}$.

A *disk graph* is the intersection graph of a family of (closed) disks in $\mathbb{R}^2$. A *unit disk graph* is the intersection graph of a family of unit disks in $\mathbb{R}^2$. It is easy to see that any (unit) disk graph is the intersection graph of a family of (unit) disks in $\mathbb{R}^2$ with the additional property that every two disks that share a point also share an interior point. That is, no two disks touch only on the boundary. Whenever we consider the geometric model of a (unit) disk graph, we will assume that it has this property.

The notion of a map graph is due to Chen et al. (1998). A *map* $\mathcal{M}$ is a pair $(\mathscr{E}, \omega)$, where $\mathscr{E}$ is a plane graph and each connected component of $\mathscr{E}$ is biconnected, and ω is a function that maps each face f of $\mathscr{E}$ to 0 or 1 in a way that whenever $\omega(f) = 1$, f is a cyclic face. A face f of $\mathscr{E}$ is called *nation* if $\omega(f) = 1$; otherwise, it is called *lake*. The graph associated with $\mathcal{M}$ is a simple graph G, where $V(G)$ consists of the nations of $\mathcal{M}$ and $E(G)$ consists of all $f_1 f_2$ such that f_1 and f_2 are adjacent (i.e., they share at least one vertex). We call G a *map graph*. By $N(\mathscr{E})$, we denote the set of nations of $\mathscr{E}$.

Treewidth. A *tree decomposition* of a graph G is a pair $(\mathcal{X}, T)$, where T is a tree and $\mathcal{X} = \{X_i \mid i \in V(T)\}$ is a collection of subsets of V such that the following conditions are satisfied.

(1) $\bigcup_{i \in V(T)} X_i = V(G)$.
(2) For each edge $xy \in (G)$, $\{x, y\} \subseteq X_i$ for some $i \in V(T)$.
(3) For each $x \in V(G)$, the set $\{i \mid x \in X_i\}$ induces a connected subtree of T.

The *width* of the tree decomposition is $\max_{i \in V(T)} |X_i| - 1$. The *treewidth* of a graph G, $\mathbf{tw}(G)$, is the minimum width over all tree decompositions of G.

We will use the following easy fact about treewidth.

Proposition 1. *The treewidth of a graph is the maximum treewidth of its connected components.*

Separators and separations. Let G be a graph, $Q \subseteq V(G)$, and let $A_1, A_2 \subseteq V(G)$ such that $A_1 \cup A_2 = V(G)$. We say that the pair (A_1, A_2) is a *separation* of G if there is no edge with one endpoint in $A_1 \setminus A_2$ and the other in $A_2 \setminus A_1$. The *order* of a separation (A_1, A_2) is $|A_1 \cap A_2|$. For a vertex subset $Q \subseteq V(G)$, we say that a separation (A_1, A_2) is a $2/3$-*balanced separation* of (G, Q) if each of the parts $A_1 \setminus A_2$ and $A_2 \setminus A_1$ contains at most $\frac{2}{3}|Q|$ vertices of Q.

The following separation property of graphs of small treewidth is well known (see, e.g., Cygan et al. (2015, Lemma 7.20)).

Proposition 2. *Let G be a graph and let $S \subseteq V(G)$. There is a $2/3$-balanced separation (A_1, A_2) of (G, S) of order at most* $\mathbf{tw}(G) + 1$.

Grids and triangulated grids. Given a positive integer t, we denote by $\boxplus_t$ the $t \times t$ grid. Formally, for a positive integer t, a $t \times t$ *grid* $\boxplus_t$ is a graph with vertex set $\{(x, y) : x, y \in \{1, \ldots, t\}\}$. Thus, $\boxplus_t$ has exactly t^2 vertices. Two different vertices (x, y) and (x', y') are adjacent if and only if $|x - x'| + |y - y'| = 1$.

For an integer $t > 0$, the graph Γ_t is obtained from the grid $\boxplus_t$ by adding, for all $1 \leq x, y \leq t - 1$, the edge $(x + 1, y), (x, y + 1)$, and making vertex (t, t) adjacent to all the other vertices (x, y) with $x \in \{1, t\}$ or $y \in \{1, t\}$, i.e., to the whole border of $\boxplus_t$. Graph Γ_9 is shown in Figure 1.

We also need the following result of Robertson and Seymour (1984).

Proposition 3. *For every $t > 1$,* $\mathbf{tw}(\boxplus_t) = t$.

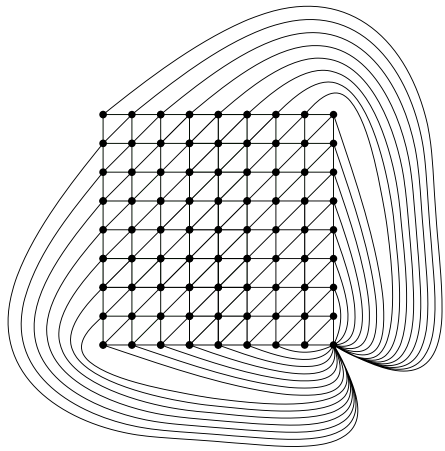

Fig. 1. Graph Γ_9.

3 OPTIMIZATION PROBLEMS AND BIDIMENSIONALITY

Our results concern graph optimization problems in which the objective is to find a vertex or edge set of minimum or maximum size that satisfies a feasibility constraint. The problems for which the task is to find a set of minimum size we call *minimization*; the problems for which the task is to find a set of maximum size we call *maximization.* A problem Π on graphs is defined by a predicate $\phi_\Pi(G,S)$ (also often called a *property*), which for a graph G and a vertex (edge) subset S of G returns **true** if S is feasible and **false** otherwise. The interpretation is that ϕ defines the space of *feasible solutions* S for a graph G by returning whether S is feasible for G. Depending on whether S is a vertex or edge subset of G, we refer to Π as a vertex or edge subset problem correspondingly.

For an example, Vertex Cover is the problem of finding a minimum vertex cover in a graph. Then, $\phi_\Pi(G,S)$ is true if and only if S is a vertex cover of G, i.e., every edge has at least one endpoint in S. For the Dominating Set problem, we put $\phi(G,S) = \textbf{true}$ if and only if $N[S] = V(G)$. Both problems are vertex subset problems.

Let us remark that there are many vertex/edge subset problems that at first glance do not look as if they could be captured by this definition. For example, the Max Leaf Spanning Tree problem, in which we are given a connected graph G and asked to find a spanning tree T of G with the maximum number of leaves, is a vertex subset problem. The reason is that G has a spanning tree with k leaves if and only if there is a set $S \subseteq V(G)$ of size at least k and $\phi(G,S)$ is true, where $\phi(G,S)$ is defined as follows:

$$
\begin{aligned}
\phi(G,S) = & \exists \text{ spanning subgraph } T \text{ of } G \text{ such that} \\
& \bullet \text{ every vertex of } S \text{ is of degree 1 in } T.
\end{aligned}
$$

Another example is the Cycle Packing problem. Here, input is a graph G and the task is to find the maximum number of pairwise vertex-disjoint cycles $C_1, C_2, \ldots, C_k$. This is again a vertex subset problem because G has k vertex-disjoint cycles if and only if there exists a set $S \subseteq V(G)$ of size at least k and $\phi(G,S)$ is true, where $\phi(G,S)$ is defined as follows:

$$
\begin{aligned}
\phi(G,S) = & \exists \text{ subgraph } G' \text{ of } G \text{ such that} \\
& \bullet \text{ each connected component of } G' \text{ is a cycle,} \\
& \bullet \text{ and each connected component of } G' \text{ contains exactly} \\
& \quad \text{one vertex of } S.
\end{aligned}
$$

Let us remark that, as in the last example, it can be that checking whether $\phi(G,S)$ is true for a given graph G and set S is NP-complete. Nevertheless, defining Cycle Packing as a vertex subset problem will allow us to obtain an EPTAS for this problem on various graph classes.

Definition 1 (OPT_Π and SOL_Π). For a vertex/edge subset minimization problem Π, we define

$$\mathrm{OPT}_\Pi(G) = \min\{|S| : \phi(G,S) = \textbf{true}\}.$$

If no S such that $\phi(G,S) = \textbf{true}$ exists, $\mathrm{OPT}_\Pi(G)$ is $+\infty$. For a vertex/edge subset maximization problem Π,

$$\mathrm{OPT}_\Pi(G) = \max\{|S| : \phi(G,S) = \textbf{true}\}.$$

In this case, if no S such that $\phi(G,S) = \textbf{true}$ exists, $\mathrm{OPT}_\Pi(G)$ returns $-\infty$. We define $\mathrm{SOL}_\Pi(G)$ to be a function that, given as an input, a graph G returns a set S of size $\mathrm{OPT}_\Pi(G)$ such that $\phi(G,S) = \textbf{true}$ and returns **null** if no such set S exists.

Bidimensional problems. For many problems, it holds that contracting an edge cannot increase the size of the optimal solution. We will say that such problems are contraction closed. For example, Dominating Set, which is the problem of finding a minimum vertex set S in a graph that dominates all vertices of $V(G) \setminus S$, is contraction closed because contracting an edge in a graph does not increase the minimum size of its dominating set. Max Leaf Spanning Tree, which is a maximization problem, is also contraction closed: if G/e has a spanning tree with at least k leaves, so does G. Formally, we have the following definition.

Definition 2 (Contraction-closed problem). A vertex/edge subset problem Π is *contraction closed* if, for any G and $e \in E(G)$, $\mathrm{OPT}_\Pi(G/e) \leq \mathrm{OPT}_\Pi(G)$.

If contracting edges, deleting edges, and deleting vertices cannot increase the size of the optimal solution, we say that the problem is minor closed. For example, Vertex Cover and Cycle Packing are minor closed.

Definition 3 (Minor-closed problem). A vertex/edge subset problem Π is *minor closed* if, for any G, edge $e \in E(G)$ and vertex $w \in V(G)$, $\mathrm{OPT}_\Pi(G/e) \leq \mathrm{OPT}_\Pi(G)$, $\mathrm{OPT}_\Pi(G-e) \leq \mathrm{OPT}_\Pi(G)$, and $\mathrm{OPT}_\Pi(G-w) \leq \mathrm{OPT}_\Pi(G)$.

The notion of a bidimensional problem was introduced by Demaine et al. (2005a). Our definition of contraction-bidimensional problems follows Fomin et al. (2011); see also Cygan et al. (2015, Chapter 7) for an introduction to bidimensionality theory.

Definition 4 (Bidimensional problem). A vertex/edge subset problem Π is

- *contraction bidimensional* if it is contraction closed and there exists a constant $\beta > 0$ such that $\mathrm{OPT}_\Pi(\Gamma_k) \geq \beta k^2$.
- *minor bidimensional* if it is minor closed and there exists a constant $\beta > 0$ such that $\mathrm{OPT}_\Pi(\boxplus_k) \geq \beta k^2$.

It is usually quite easy to determine whether a problem is contraction (or minor) bidimensional. Take for an example Independent Set. Contracting an edge may never increase the size of the maximum independent set; thus, the problem is contraction closed. Furthermore, in Γ_k, the vertex set

$$\{(x,y) : x \equiv k+1 \mod 2 \textit{and}\, y \equiv k+1 \mod 2\}$$

forms an independent set of size $\frac{(k-1)^2}{4}$. Thus, Independent Set is contraction bidimensional. On the other hand, deleting edges may increase the size of a maximum size independent set in G. Thus, Independent Set is not minor bidimensional.

Separability. The notion of separable problems was introduced by Demaine and Hajiaghayi (2005). Informally, separable problems Π are well behaved in the sense that whenever we have a small separator in the graph that splits the graph in two parts L and R, the intersection $|X \cap L|$ of L with any optimal solution X to the entire graph is a good estimate of $OPT_{\Pi}(G[L])$. We provide many examples of separable problems in Section 4.

Separability allows us to prove decomposition theorems that are very useful for deriving EPTASs. Similar decomposition theorems may also be used to obtain polynomial kernels (see Fomin et al. (2010)).

Definition 5 (Separability) (Demaine and Hajiaghayi 2005). Let $f : \mathbb{Z}^+ \to \mathbb{Z}^+$ be a function. We say that a vertex/edge subset problem Π is f *separable* if, for any graph G, subset $L \subseteq V(G)$, and function SOL_{Π}, it holds that

$$|\,\mathrm{SOL}_{\Pi}(G) \cap L| - f(t) \leq \mathrm{OPT}_{\Pi}(G[L]) \leq |\,\mathrm{SOL}_{\Pi}(G) \cap L| + f(t),$$

where $t = |\partial_G(L)|$. Π is called *separable* if there exists a function f such that Π is f separable. Π is called *linear separable* if there exists a constant c such that Π is $c \cdot t$ separable.

CMSO-definable problems. The syntax of Monadic Second Order Logic (MSO) of graphs includes the logical connectives $\vee$, $\wedge$, $\neg$, $\Leftrightarrow$, $\Rightarrow$, variables for vertices, edges, sets of vertices, and sets of edges, the quantifiers $\forall$, $\exists$ that can be applied to these variables, and the following five binary relations:

(1) $u \in U$, where u is a vertex variable and U is a vertex set variable;
(2) $d \in D$, where d is an edge variable and D is an edge set variable;
(3) $\mathbf{inc}(d, u)$, where d is an edge variable, u is a vertex variable, and the interpretation is that the edge d is incident with the vertex u;
(4) $\mathbf{adj}(u, v)$, where u and v are vertex variables and the interpretation is that u and v are adjacent;
(5) equality of variables representing vertices, edges, sets of vertices, and sets of edges.

In addition to the usual features of MSO, if we have atomic sentences testing whether the cardinality of a set is equal to q modulo r, where q and r are integers such that $0 \leq q < r$ and $r \geq 2$, then this extension of MSO is called *counting monadic second-order logic (CMSO)*. Thus, CMSO is MSO enriched with the following atomic sentence for a set S:

$$\mathbf{card}_{q,r}(S) = \mathbf{true} \text{ if and only if } |S| \equiv q \pmod{r}.$$

Refer to Arnborg et al. (1991) and Courcelle (1990, 1997) for a detailed introduction on CMSO.

We consider CMSO sentences evaluated either on graphs or on annotated graphs. By *annotated graph*, we mean a pair (G, S), where S is a subset of the vertices or edges of a graph G, i.e., S contains the *annotated* vertices or edges of G. In optimization problems on annotated graphs, annotated set S is used to specify additional constraints or relaxations of the solution. For example, a solution can be a subset or a superset of annotated set S or, as in annotated Dominating Set, annotated vertices are not required to be dominated by the solution, and so on. We say that a CMSO-formula φ *expresses* an (annotated) graph property $\phi_{\Pi}(G, S)$ if $\phi_{\Pi}(G, S)$ is true if and only if (G, S) models φ (i.e., the formula φ is true exactly on (annotated) graphs G and vertex/edge subsets S such that $\phi_{\Pi}(G, S)$ is true).

Min-CMSO and Max-CMSO problems are graph optimization problems for which the objective is to find a maximum- or minimum-sized vertex or edge set satisfying a CMSO-expressible property. In other words, a vertex/edge subset minimization (or maximization) problem with feasibility function ϕ is a Min-CMSO problem (or Max-CMSO problem) if there exists a CMSO sentence ψ

such that $\phi(G, S) = \textbf{true}$ if and only if $(G, S) \models \psi$. Thus, in a Min/Max-CMSO graph problem Π, we are given a graph G as input. The objective is to find a minimum/maximum cardinality vertex vertex/edge set S such that the CMSO-expressible predicate $\phi_\Pi(G, S)$ is satisfied.

Reducibility and η-modulated problems. Reducibility is the central notion of this article. The intuition behind the notion of reducibility is the following. Being reducible allows us to "sacrifice" a set of vertices X (e.g., deleting them or putting in a solution) by creating a new problem whose solution and treewidth are "approximated" by the original solution and treewidth, and such that from the solution of the new problem, one can in polynomial time reconstruct an approximate solution of the original problem.

Definition 6 (Reducibility). A graph optimization problem Π defined by a predicate $\phi_\Pi(G, S)$ is *reducible* if there exists a Min/Max-CMSO problem Π' with CMSO-expressible property $\phi_{\Pi'}$, a function $f : \mathbb{N} \to \mathbb{N}$, and a constant ρ_Π such that

(1) there is a polynomial-time algorithm that, given graph G and set $X \subseteq V(G)$, outputs graph G' such that $\mathbf{tw}(G') \leq f(\mathbf{tw}(G - X))$ and $|\,\mathrm{OPT}_{\Pi'}(G') - \mathrm{OPT}_\Pi(G)| \leq \rho_\Pi \cdot |X|$; and
(2) there is a polynomial-time algorithm that, given graph G and $X \subseteq V(G)$, graph G' and a vertex (edge) set $S' \subseteq V(G')$ ($S' \subseteq E(G')$) such that $\phi_{\Pi'}(G', S')$ holds, outputs $S \subseteq V(G)$ such that $\phi_\Pi(G, S) = \textbf{true}$ and $||S| - |S'|| \leq \rho_\Pi \cdot |X|$.

In most of the cases, problem Π' will be an "annotated" version of the optimization problem, in which for a set of annotated vertices or edges, we require special conditions. For example, to show that Dominating Set is reducible, for vertex set X of G, we put $G' = G - X$. Then, problem Π' defined on G' is the variant of domination in which we annotate the vertices of G' adjacent (in G) to X as not required to be dominated by a solution in G'. In other words, we are looking for a set of vertices in G' that dominates all but the annotated vertices. Such an annotated domination is a Min/Max-CMSO problem and it is easy to check that both properties of the definition hold. Thus, Π is reducible. In Section 4, we give more examples of reducible problems.

Definition 7 (Treewidth η-modulated). For a nonnegative integer η, a graph optimization problem Π is called *treewidth η-modulated*, or simply, *η-modulated*, if there is a polynomial-time algorithm that, given a graph G, outputs a set X of size $O(\mathrm{OPT}_\Pi(G))$ such that $\mathbf{tw}(G - X) \leq \eta$.

For example, Vertex Cover is η-modulated for $\eta = 0$: there is a polynomial time algorithm computing a vertex cover X of size at most $2 \cdot \mathrm{OPT}_\Pi(G)$ (Nemhauser and Trotter 1974). Since graph $G - X$ has no edges, its treewidth is 0.

Subquadratic grid minor property. We will build EPTASs on graphs with specific properties. In general, it is known that there exists a constant c such that any graph G that excludes $\boxplus_t$ as a minor has treewidth at most $O(t^c)$. The exact value of c remains unknown, but it is at least 2 and at most 19 (Chuzhoy 2016; Chekuri and Chuzhoy 2016). We will restrict our attention to graph classes with $c < 2$.

Definition 8 (SQGM property). We say that a graph class $\mathcal{G}$ has the *subquadratic grid minor (SQGM) property* if there exist constants $\alpha > 0$ and $1 \leq c < 2$ such that, for any $t > 0$, every graph $G \in \mathcal{G}$, excluding $\boxplus_t$ as a minor, has treewidth at most $\alpha \cdot t^c$. In the cases for which we need to specify the parameter c, we say that graph class $\mathcal{G}$ has the SQGM property with parameter c.

Problems that are contraction closed but not minor closed are considered on more restricted classes of graphs.

Definition 9 (SQGC property). We say that a graph class $\mathcal{G}$ has the *subquadratic gamma contraction (SQGC) property* if there exist constants $\alpha > 0$ and $1 \le c < 2$ such that for any $t > 0$, every connected graph $G \in \mathcal{G}$, excluding Γ_t as a contraction, has treewidth at most $\alpha \cdot t^c$.

Since Γ_t contains $\boxplus_t$ as a minor (in fact, as a subgraph), we obtain the following observation.

Observation 1. *Every graph class $\mathcal{G}$ with the SQGC property has the SQGM property.*

The following proposition follows directly from the theorem on the linearity of excluded grid-minor in H-minor-free graphs proven by Demaine and Hajiaghayi (2008c) and its analogue for Γ-contractions from Fomin et al. (2011).

Proposition 4. *For every graph H, any H-minor-free graph class $\mathcal{G}$ has the SQGM property with parameter $c = 1$. If H is an apex graph, then $\mathcal{G}$ has the SQGC property with $c = 1$.*

Since every planar graph excludes graphs K_5 or $K_{3,3}$ as a minor, the class of planar graphs has the SQGM property. More generally, the class of graphs of bounded genus also has the SQGM property. By the result of Eppstein (2000), every minor-closed class of graphs of bounded local treewidth (like the classes of planar graphs or graphs of bounded genus) exclude some fixed apex graph as a minor. Thus, these classes of graphs have SQGC property.

The bidimensionality properties of problems and SQGM properties of graph classes allow us to establish *parameter-treewidth bounds*—a tight dependence between the size of the optimal solution and the treewidth of the input graph. This relationship was first observed by Demaine et al. (2005a). The bound for contraction-bidimensional problems presented here is essentially identical to the one presented in Fomin et al. (2011). We reprove the lemmata here because of slight differences in definitions.

Lemma 1. *For any minor-bidimensional problem Π on a graph class $\mathcal{G}$ with the SQGM property with the parameter $c < 2$, there exists constant $\gamma > 0$ such that, for any graph $G \in \mathcal{G}$, $\mathbf{tw}(G) \le \gamma \cdot (\mathrm{OPT}_\Pi(G))^\epsilon$, where $\epsilon = c/2$.*

Proof. Let α and c be the constants from the definition of the SQGM property. Then, any graph $G \in \mathcal{G}$ that excludes a $\boxplus_t$ as a minor has treewidth at most αt^c. Let β be the constant from the definition of minor bidimensionality of Π, i.e, $\mathrm{OPT}_\Pi(\boxplus_t) \ge \beta t^2$. Consider now a graph $G \in \mathcal{G}$. Let t be the maximum integer such that G contains $\boxplus_t$ as a minor. We have that $\mathbf{tw}(G) < \alpha(t+1)^c$. Rearranging terms yields that $(\frac{\mathbf{tw}(G)}{\alpha})^{1/c} < t+1$, implying that $t \ge (\frac{\mathbf{tw}(G)}{\alpha})^{1/c}$. Since Π is minor closed, it follows that $\mathrm{OPT}_\Pi(G) \ge \mathrm{OPT}_\Pi(\boxplus_t)$, and since Π is minor bidimensional, we have that

$$\mathrm{OPT}_\Pi(G) \ge \mathrm{OPT}_\Pi(\boxplus_t) \ge \beta t^2 \ge \beta \left(\frac{\mathbf{tw}(G)}{\alpha}\right)^{2/c}.$$

Hence,

$$\mathbf{tw}(G) \le \frac{\alpha}{\beta^{\frac{c}{2}}} \cdot \mathrm{OPT}_\Pi(G)^{\frac{c}{2}}.$$

Since $c < 2$ in the definition of the SQGM property, the statement of the lemma follows. □

Lemma 2. *For any contraction-bidimensional problem Π on a graph class $\mathcal{G}$ with the SQGC property with the parameter $c < 2$, there exists constant $\gamma > 0$ such that, for any connected graph $G \in \mathcal{G}$, $\mathbf{tw}(G) \le \gamma \cdot (\mathrm{OPT}_\Pi(G))^\epsilon$, where $\epsilon = c/2$.*

The proof of Lemma 2 is almost identical to the proof of Lemma 1 except for the following difference. Lemma 1 is for minor-closed problems, on graph classes $\mathcal{G}$ with the SQGM property and works for *every* $G \in \mathcal{G}$. Lemma 2 is for contraction-closed problems, on graph classes $\mathcal{G}$ with the SQGC property and works for *connected* graphs $G \in \mathcal{G}$.

However, when a problem is contraction bidimensional and *separable*, then it is possible to extend Lemma 2 to disconnected graphs, since the issue of different connected components influencing the value of the optimal solution disappears.

LEMMA 3. *For any contraction-bidimensional separable problem Π on a graph class $\mathcal{G}$ with the SQGC property, there exist constants $0 < \gamma$, $0 < \epsilon < 1$ such that, for any graph $G \in \mathcal{G}$, $\mathbf{tw}(G) \leq \gamma \cdot (OPT_{\Pi}(G))^{\epsilon}$.*

PROOF. Since Π is separable, there exists a constant d such that, for any graph G and connected component C of G, it holds that

$$\mathrm{OPT}_{\Pi}(G[C]) \leq |\,\mathrm{SOL}_{\Pi}(G) \cap C| + d \leq \mathrm{OPT}_{\Pi}(G) + d.$$

By Lemma 2, there exist constants γ and $\epsilon < 1$ such that $\mathbf{tw}(G[C]) \leq \gamma \cdot \mathrm{OPT}_{\Pi}(G[C])^{\epsilon}$ for every component C. Since the treewidth of G is at most the maximum of the treewidth of its connected components, the lemma follows by increasing γ by a factor d. □

Treewidth modulator. We say that a set $S \subseteq V(G)$ is a *treewidth-η-modulator* if $\mathbf{tw}(G - S) \leq \eta$. For every $\eta \geq 0$, we define the following problem.

TREEWIDTH-η MODULATOR	
Instance:	A graph G.
Objective:	Find a treewidth-η-modulator of minimum size.

Since the graph is of treewidth 0 if and only if it has no edges, the VERTEX COVER problem is TREEWIDTH-η MODULATOR for $\eta = 0$. The treewidth of a graph G is at most 1 if and only if G is a forest. Hence, the FEEDBACK VERTEX SET problem is TREEWIDTH-η MODULATOR for $\eta = 1$.

It is easy to see that the TREEWIDTH-η MODULATOR problem is minor closed. By Proposition 3, every $(\eta + 1) \times (\eta + 1)$ subgrid of $\boxplus_t$ must contain at least one vertex of any solution. Therefore,

$$\mathrm{OPT}_{\Pi}(\boxplus_t) \geq \left\lfloor \left(\frac{t}{\eta + 1}\right) \right\rfloor^2.$$

Thus, TREEWIDTH-η MODULATOR is minor bidimensional.

By Lemma 1, the bidimensionality of TREEWIDTH-η MODULATOR yields the following lemma.

LEMMA 4. *For every graph class $\mathcal{G}$ with the SQGM property and every $\eta \geq 0$, there exist constants $\beta \geq 0$ and $0 \leq \lambda < 1$ such that any graph $G \in \mathcal{G}$ with a treewidth-η-modulator S has treewidth at most $\beta \cdot |S|^{\lambda}$.*

Note that when $\mathcal{G}$ has the SQGM property with the parameter c, then in Lemma 4, $\lambda = c/2$.

4 REDUCIBLE BIDIMENSIONAL LINEAR-SEPARABLE PROBLEMS

In this section, we give examples of problems that are reducible, bidimensional, and linear separable. As we will see later in Lemma 8, the combination of these conditions with SQGM or SQGC properties yields that these problems are treewidth η-modulated. This, in turn, by Theorem 2, guarantees an EPTAS for these problems.

4.1 Domination, Independence, and Connectivity Problems

In the r-DOMINATING SET problem, we are given a graph G; the objective is to find a minimum-size subset $S \subseteq V(G)$ such that every vertex outside S is within distance at most r from some vertex of S. In other words, $B^r_G(S) = V(G)$. For $r = 1$, this is the classical DOMINATING SET problem. If we also demand $G[S \cap C]$ to be connected for every connected component C of graph G, we obtain the CONNECTED DOMINATING SET problem. In the CONNECTED VERTEX COVER problem, we are given a graph G and the objective is to find a minimum-size subset $S \subseteq V(G)$ such that S is a vertex cover of G, that is, every edge in $E(G)$ has at least one endpoint in S and such that for every connected component C of G, $G[S \cap C]$ is connected. It is known that r-Dominating Set, Connected Dominating Set, and Connected Vertex Cover are contraction bidimensional (Demaine et al. 2005a).

We show that $\Pi = r$-Dominating Set is linear separable. For graph G, let $L \subseteq V(G)$ with $|\partial(L)| \leq t$. Observe that the set $(\mathrm{SOL}_\Pi(G) \cap L) \cup \partial(L)$ is r-dominating in $G[L]$. Indeed, every vertex of $v \in L$ is either within distance at most r from some vertex from $\mathrm{SOL}_\Pi(G) \cap L$, or from a vertex $\mathrm{SOL}_\Pi(G) \setminus L$. In each of the cases, each of the paths of length at most r from $\mathrm{SOL}_\Pi(G)$ to v is either entirely in L or contains a vertex of $\partial(L)$. Thus,

$$\mathrm{OPT}_\Pi(G[L]) \leq |\,\mathrm{SOL}_\Pi(G) \cap L| + t.$$

On the other hand, if $|\,\mathrm{SOL}_\Pi(G) \cap L| - t > \mathrm{OPT}_\Pi(G[L])$, then set $\mathrm{SOL}_\Pi(G[L]) \cup \partial(L) \cup \mathrm{SOL}_\Pi(G - L)$ is r-dominating in G and is of size less than $\mathrm{OPT}_\Pi(G)$, which is a contradiction. Hence,

$$|\,\mathrm{SOL}_\Pi(G) \cap L| - t \leq \mathrm{OPT}_\Pi(G[L]),$$

and r-Dominating Set is linear separable.

We now show that r-Dominating Set is reducible. For a given graph G and set X, let $G' = G - X$ and let $R = B^r_G(X) \setminus X$. Then, $\mathbf{tw}(G') = \mathbf{tw}(G - X)$. Problem Π' is the following annotated version of r-Dominating Set. In graph G', we annotate the vertex set R. We want to find a set $S' \subseteq V(G')$ of minimum cardinality such that every vertex in $V(G') \setminus (S' \cup R)$ is within distance at most r from a vertex in S'. In other words, the set S' r-dominates all vertices of G' except annotated vertices R. It is easy to show that Π' is a Min-CMSO problem. Note that, for any r-dominating set S of G, $S \setminus X$ is a feasible solution to Π' on G'. Conversely, for any feasible solution S' of Π' on G', we have that $S' \cup X$ is an r-dominating set of G. Hence, r-Dominating Set is reducible.

Proofs that Connected Dominating Set and Connected Vertex Cover are linear separable are similar to the proof for r-Dominating Set. For example, for Connected Dominating Set, let L be a vertex subset of G. Let us assume first that $G[L]$ is connected. If $\partial(L) = \emptyset$, then separability follows trivially. If $\partial(L) \neq \emptyset$, we first augment $SOL_\Pi(G) \cap L$ by adding $\partial(L)$ to it. The set $Q = (SOL_\Pi(G) \cap L) \cup \partial(L)$ is a dominating set of $G[L]$. Since $\partial(L) \neq \emptyset$, we have that every connected component of $G[Q]$ contains at least one vertex of $\partial(L)$. (Otherwise, $SOL_\Pi(G)$ is either not connected or not dominating in G.) Hence, set Q contains at most $|\partial L|$ connected components and can be turned into a connected set by adding at most $2|\partial L| - 1$ vertices. When $G[L]$ is not connected, we apply the same construction for each of its connected components C and for each of the sets $C \cap (SOL_\Pi(G) \cup \partial(L))$.

We now prove that Connected Dominating Set is reducible. Given a graph G and set X, let $G' = G - X$ and let $R = N(X)$. The annotated problem Π' is to find a minimum-sized set $S' \subseteq V(G')$ such that every vertex in $V(G') \setminus (S' \cup R)$ has a neighbor in S' and every connected component of $G'[S']$ contains a vertex in R. Note that for every connected dominating set S of G, $S \setminus X$ is a feasible solution to Π' on G'. Conversely, for any feasible solution S' of Π' on G', we have that $S = S' \cup X$ is a dominating set of G and has at most $|X|$ connected components. Since S is a dominating set, it is sufficient to add $2(|X| - 1)$ vertices to S in order to turn it into a connected dominating set of G. Hence, Connected Dominating Set is reducible. The proof that Connected Vertex Cover is reducible is identical.

In the r-Scattered Set problem, the task is for a given graph G to find a maximum set of vertices $S \subseteq V(G)$ such that the distance between any two vertices of S in G is more than r. For $r = 1$, this is the Independent Set problem. The proof that r-Scattered Set is contraction bidimensional, linear separable, and reducible is similar to the one for r-Dominating Set and is omitted. We collect all of the above observations in the following lemma.

Lemma 5. *r-Dominating Set, Connected Dominating Set, Connected Vertex Cover, and r-Scattered Set are contraction bidimensional, linear separable, and reducible.*

4.2 Covering and Packing Problems

Minor covering and packing. This section presents a few generic problems. Each of the problems subsumes many problems in itself and fits in our framework. Let $\mathcal{F}$ be a finite set of connected graphs containing at least one planar graph. We define the following problem.

> $\mathcal{F}$-Deletion
> *Instance:* A graph G.
> *Objective:* Find a set $S \subseteq V(G)$ of minimum size such that $G - S$ contains no graph from $\mathcal{F}$ as a minor.

We consider the situation when set $\mathcal{F}$ contains at least one planar graph because, as we will see later, in this case, $\mathcal{F}$-Deletion is bidimensional. However, even in this special scenario, $\mathcal{F}$-Deletion generalizes many interesting problems. For example,

— Vertex Cover is the case when $\mathcal{F} = \{K_2\}$. Here, K_i is a complete graph on i vertices.
— When $\mathcal{F} = \{K_3\}$, this is the Feedback Vertex Set problem.
— Diamond Hitting Set is the case when $\mathcal{F} = \{K_4\}$.
— Other choices for $\mathcal{F}$ lead to vertex deletion into outerplanar graphs, series-parallel graphs, and graphs of constant treewidth (Treewidth-η Modulator) or pathwidth.

$\mathcal{F}$-Deletion can be seen as a variant of the Hitting Set problem, in which the task is to "hit" all forbidden minors. The dual maximization problem is the following.

> $\mathcal{F}$-Packing
> *Instance:* A graph G.
> *Objective:* Find a maximum size collection of vertex disjoint subgraphs such that each of them contains a graph from $\mathcal{F}$ as a minor.

In particular, $\mathcal{F}$-Packing contains problems such as Cycle Packing as a special case.

It is easy to see that both $\mathcal{F}$-Deletion and $\mathcal{F}$-Packing are minor-closed problems. Since we assume that set $\mathcal{F}$ contains at least one planar graph, we can select a smallest planar graph F in $\mathcal{F}$. By the result of Robertson et al. (1994), F is a minor of the $t \times t$ grid $\boxplus_t$, where $t = 14|V(F)| - 24$. Grid $\boxplus_r$ contains r^2/t^2 disjoint subgrids, each containing F as a minor. Since every solution of $\mathcal{F}$-Deletion should contain at least one vertex of each of the r^2/t^2 subgrids, we have that $\mathcal{F}$-Deletion is minor bidimensional. Similarly, $\boxplus_r$ contains at least r^2/t^2 vertex-disjoint subgraphs, each containing F as a minor; hence, $\mathcal{F}$-Packing is minor bidimensional.

We now prove that $\Pi = \mathcal{F}$-Deletion is linear separable. For graph G, let $L \subseteq V(G)$ with $|\partial(L)| \leq t$. Then, $(\text{SOL}_\Pi(G) \cap L) \cup \partial(L)$ "hits" every subgraph of $G[L]$ containing a graph from $\mathcal{F}$ as a minor. Thus, $\text{OPT}_\Pi(G[L]) \leq |\text{SOL}_\Pi(G) \cap L| + t$. To prove that $|\text{SOL}_\Pi(G) \cap L| - t \leq \text{OPT}_\Pi(G[L])$, observe that if $|\text{SOL}_\Pi(G) \cap L| - t > \text{OPT}_\Pi(G[L])$, then set $\text{SOL}_\Pi(G[L]) \cup \partial(L) \cup \text{SOL}_\Pi(G - L)$ is a set hitting all minors from $\mathcal{F}$ in G of size smaller than $\text{OPT}_\Pi(G)$, which is a contradiction. Hence, $\mathcal{F}$-Deletion is linear separable.

The proof of separability of $\mathcal{F}$-Packing goes along the same lines as the proof for $\mathcal{F}$-Deletion, but with a few notable differences. We view $\mathcal{F}$-Packing as a problem of finding a maximum vertex set S such that there is a subgraph H of G, such that every connected component of H contains exactly one vertex of S and contains some graph from $\mathcal{F}$ as a minor. Then, the main observation implying linear separability is that by deleting t vertices from G, we cannot touch more than t components from H.

Finally, it is easy to see that both $\mathcal{F}$-Deletion and $\mathcal{F}$-Packing are reducible. Given G and X, we let $G' = G - X$. For $\mathcal{F}$-Deletion, X can be added to an optimal solution in G' at the cost of $|X|$. For $\mathcal{F}$-Packing, at most $|X|$ of the minors of graphs in $\mathcal{H}$ contain a vertex in X and got removed when X was deleted. Expressing both problems as Min/Max-CMSO problems is routine.

Lemma 6. *Let $\mathcal{F}$ be a finite set of graphs containing a planar graph. Then, $\mathcal{F}$-Deletion and $\mathcal{F}$-Packing are minor bidimensional, linear separable, and reducible.*

5 SCALING LEMMA

In this section, we prove the following lemma, which is crucial in our analysis. Informally, the lemma says the following. Let X be a treewidth-η-modulator of a graph G from graph class $\mathcal{G}$ with the SQGM property. Then, for any $\epsilon > 0$, one can scale down in polynomial time set X to set X' of size $\epsilon \cdot |X|$ such that every connected component C of $G - X'$ is separated from the remaining graph by a constant number of vertices and contains only a constant number of vertices from X. Since X is also a treewidth-η-modulator in $G[C]$, this implies, in particular, that the treewidth of C—and thus of $\mathbf{tw}(G - X')$—is bounded by a constant depending on ϵ, η, and class $\mathcal{G}$ only. Thus, the lemma allows us to obtain a smaller treewidth-modulator from a given one.

Lemma 7 (Scaling Lemma). *Let $\mathcal{G}$ be a hereditary graph class with the SQGM property. For any $1 > \varepsilon > 0$ and $\eta > 0$, there is γ such that for any $G \in \mathcal{G}$ and treewidth-η-modulator X of G, there is $X' \subseteq V(G)$ such that*

- *$|X'| \leq \varepsilon |X|$, and*
- *for every connected component C of $G - X'$, we have that $|C \cap X| \leq \gamma$ and $|N(C)| \leq \gamma$.*

Moreover, such a set X' can be computed from G and X in polynomial time, where the polynomial is independent of ε and η.

Proof. Since $\mathcal{G}$ is a hereditary graph class with the SQGM property, by Lemma 4, there exist constants β and $0 < \lambda < 1$ such that, for every $G \in \mathcal{G}$ such that G has a treewidth-η-modulator of size at most k, we have that $\mathbf{tw}(G) \leq \beta \cdot k^{\lambda}$. We select the constant γ based on λ, β, η, and ε. Let

$$\rho = \min_{1/3 \leq \alpha \leq 2/3} \alpha^{\lambda} + (1 - \alpha)^{\lambda}.$$

Observe that since $0 < \lambda < 1$, for any $a > 0$, $b > 0$, we have that $a^{\lambda} + b^{\lambda} > (a + b)^{\lambda}$. Hence, $\rho > 1$. We also define

$$\delta = \frac{(2\varepsilon + 1)(\beta + \eta + 1)}{\rho - 1},$$

and finally

$$\gamma = \left(\frac{3\delta}{\varepsilon}\right)^{\frac{1}{1-\lambda}}.$$

The choice of these constants will become clear during the course of the proof. Let us also note that $\rho < 2$. Thus, $\delta \geq \varepsilon$; hence, $\gamma \geq 1$.

To prove that γ is the required constant, we define the value $T_{\gamma}(k)$ as the minimum size of a set X' satisfying conditions of the lemma. That is, $T_{\gamma}(k)$ is the smallest integer such that if $G \in \mathcal{G}$ and there is $X \subseteq V(G)$ with $\mathbf{tw}(G - X) \leq \eta$ and $|X| \leq k$, then there is $X' \subseteq V(G)$ of size at most $T_\gamma(k)$ such that, for every connected component C of $G - X'$, we have that $|C \cap X| \leq \gamma$ and $|N(C)| \leq \gamma$. In other words, $T_\gamma(k)$ is the minimum size of a vertex set X' such that every connected component C of $G - X'$ has at most γ neighbors in X' and contains at most γ vertices of X.

Then, to prove the combinatorial statement of the lemma, we have to show that for every $k \geq 1$,

$$T_\gamma(k) \leq \varepsilon k. \tag{1}$$

Let us observe that Equation (1) trivially holds for $k \leq \gamma$. Indeed, let X be a treewidth-η-modulator of G of size k (if G has no modulator of size k, there is nothing to prove). In this case, we put $X' = \emptyset$. Then, $0 = |X'| \leq \varepsilon k$ and for every connected component C of $G - X' = G$, we have that $|C \cap X| \leq k \leq \gamma$ and $|N(C)| = 0 \leq \gamma$. Thus, for $k \leq \gamma$, $T_\gamma(k) = 0$ and (1) holds.

In order to prove Equation (1) for $k > \gamma$, we prove a slightly stronger statement: for $k \geq \gamma/3$,

$$T_\gamma(k) \leq \varepsilon k - \delta k^\lambda. \tag{2}$$

We prove Equation (2) by induction on k. For the base case, we choose $\gamma/3 \leq k \leq \gamma$. Then, the choice of γ implies that

$$\varepsilon k - \delta k^\lambda \geq \varepsilon \frac{\gamma}{3} - \delta \gamma^\lambda \geq 0.$$

On the other hand, we already have proved that, for $k \leq \gamma$, $T_\gamma(k) = 0$. Thus, for $\gamma/3 \leq k \leq \gamma$,

$$T_\gamma(k) \leq \varepsilon k - \delta k^\lambda,$$

which concludes the base case of the induction.

For the inductive step, let $k > \gamma$. Let $G \in \mathcal{G}$ be a graph with a treewidth-η-modulator of size at most k. By Lemma 4, the treewidth of G is at most βk^λ. By Proposition 2, there is a 2/3-balanced separation of (G, X) of order at most $\mathbf{tw}(G) + 1$. Hence, there is a partition of $V(G)$ into L, S, and R such that $|S| \leq \beta k^\lambda + 1$, $N(L) \subseteq S$, $N(R) \subseteq S$, $|L \cap X| \leq 2k/3$, and $|R \cap X| \leq 2k/3$. Deleting S from the graph G yields two graphs $G[L]$ and $G[R]$ with no edges between them. Since $\mathcal{G}$ is a hereditary class of graphs, we can proceed recursively. For that, we put S into X' and then proceed recursively in $G[L \cup S]$ and $G[R \cup S]$, starting from the sets $S \cup (X \cap L)$ and $S \cup (X \cap R)$ in $G[L \cup S]$ and $G[R \cup S]$, respectively. This yields the following recurrence for T_γ:

$$T_\gamma(k) \leq \max_{1/3 \leq \alpha \leq 2/3} T(\alpha k + \beta k^\lambda + 1) + T((1-\alpha)k + \beta k^\lambda + 1) + \beta k^\lambda + 1.$$

Observe that since $k > \gamma$ and $1/3 \leq \alpha \leq 2/3$, we have that $\alpha k \geq \gamma/3$ and $(1 - \alpha k) \geq \gamma/3$. On the other hand, $\max\{\alpha k, (1-\alpha)k\} \leq 2k/3$ and $2k/3 + \beta k^\lambda + 1 < k$ for $k > \gamma$. The last inequality can be proved by observing that the function $k/3 - \beta k^\lambda - 1$ is monotonically increasing for $k > (3\lambda\beta)^{\frac{1}{1-\lambda}}$ and that $\gamma > (3\beta)^{\frac{1}{1-\lambda}} > (3\lambda\beta)^{\frac{1}{1-\lambda}}$. Then, the induction hypothesis yields the following.

$$\begin{aligned}
T_\gamma(k) &\leq \max_{1/3 \leq \alpha \leq 2/3} T(\alpha k + \beta k^\lambda + 1) + T((1-\alpha)k + \beta k^\lambda + 1) + \beta k^\lambda + 1 \\
&\leq \max_{1/3 \leq \alpha \leq 2/3} \varepsilon(k + 2\beta k^\lambda + 2) - \delta(\alpha k + \beta k^\lambda + 1)^\lambda - \delta(1-\alpha)k + \beta k^\lambda + 1)^\lambda + \beta k^\lambda + 1 \\
&\leq \max_{1/3 \leq \alpha \leq 2/3} \varepsilon k - \delta(\alpha k)^\lambda - \delta((1-\alpha)k)^\lambda + (2\varepsilon + 1)(\beta k^\lambda + 1) \\
&\leq \max_{1/3 \leq \alpha \leq 2/3} \varepsilon k - \delta k^\lambda(\alpha^\lambda + (1-\alpha)^\lambda) + (2\varepsilon + 1)(\beta k^\lambda + 1) \\
&\leq \varepsilon k - \delta k^\lambda - \delta(\rho - 1)k^\lambda + (2\varepsilon + 1)(\beta k^\lambda + 1) \leq \varepsilon k - \delta k^\lambda.
\end{aligned}$$

The last inequality holds whenever $\delta(\rho - 1)k^\lambda \geq (2\varepsilon + 1)(\beta k^\lambda + 1)$, which is ensured by the choice of δ and the fact that $k^\lambda \geq 1$. This concludes the proof of Equation (2), and thus of Equation (1).

By the definition of $T_\gamma(k)$, Equation (1) implies that there exists a set X' of size at most εk such that, for every component C of $G - X'$, we have that $|C \cap X| \leq \gamma$ and $|N(C)| \leq \gamma$.

What remains is to show that X' can be computed from G and X in polynomial time. The inductive proof can be converted into a recursive algorithm. The only computationally hard step of the proof is the construction of a tree-decomposition of G in each inductive step. Instead of computing the treewidth exactly, we use the $d^*\sqrt{\log \mathbf{tw}(G)}$-approximation algorithm by Feige et al. (2008), where d^* is a fixed constant. Thus, when we partition $V(G)$ into L, S, and R using Proposition 2, the upper bound on the size of S will be $d^*(\beta k^\lambda)\sqrt{\log(\beta k^\lambda)}$ instead of βk^λ. However, for any $\lambda < \lambda' < 1$, there is a β' such that $d^*(\beta k^\lambda)\sqrt{\log(\beta k^\lambda)} < \beta' k^{\lambda'}$. Now, we can apply the above analysis with β' instead of β and λ' instead of λ to bound the size of the set X' output by the algorithm. This concludes the proof of the lemma. □

The following corollary is a direct consequence of Lemma 7. Nevertheless, we find it worthwhile to be mentioned separately.

Corollary 1. *Let $\mathcal{G}$ be a hereditary graph class with the SQGM property with parameter 2λ. For any $\epsilon > 1$ and $\tau = \mathcal{O}((\frac{1}{\epsilon})^{\frac{\lambda}{1-\lambda}})$, we have that, for any $G \in \mathcal{G}$ and $X \subseteq V(G)$ with $\mathbf{tw}(G - X) \leq \eta$, there is $X' \subseteq V(G)$ satisfying $|X'| \leq \epsilon |X|$ such that $\mathbf{tw}(G - X') \leq \tau$.*

Proof. We apply Lemma 7 on G and X to obtain the set X' of size $\epsilon|X|$. Observe that, in the proof of Lemma 7, $\gamma = \mathcal{O}((\frac{1}{\epsilon})^{\frac{1}{1-\lambda}})$. The treewidth of $G - X'$ equals the maximum treewidth of a connected component C of $G - X'$. However, $|C \cap X| \leq \gamma$; thus, $\mathbf{tw}(G[C]) = \mathcal{O}(\gamma^\lambda)$, concluding the proof. □

6 APPROXIMATION SCHEME: PUTTING IT ALL TOGETHER

We are ready to state our first meta-theorem.

Theorem 1. *Let Π be an η-modulated and reducible graph optimization problem. Then, Π has an EPTAS on every hereditary graph class $\mathcal{G}$ with the SQGM property.*

Proof. Let $\phi(G, S)$ be a predicate defining an η-modulated and reducible graph optimization problem Π. Let G be the input to Π and $\epsilon > 0$ be a constant.

Since Π is η-modulated, there is a constant $\rho_1 > 0$ and a polynomial time algorithm that outputs a set X such that $|X| \leq \rho_1 \,\mathrm{OPT}_\Pi(G)$ and $\mathbf{tw}(G - X) \leq \eta$. Furthermore, since $\mathcal{G}$ is a hereditary graph class with the SQGM property and $\mathbf{tw}(G - X) \leq \eta$, by Lemma 4, there exist constants β and $\lambda < 1$ such that $\mathbf{tw}(G) \leq \beta|X|^\lambda$. Since problem Π defined by a predicate $\phi(G, S)$ is reducible, there exist a Min/Max-CMSO problem Π' defined by a CMSO-expressible predicate $\phi'(G, S)$, a constant ρ_Π, and a function $f : \mathbb{N} \to \mathbb{N}$ such that:

(R1) there is a polynomial-time algorithm that, given G and $X' \subseteq V(G)$, outputs G' such that $|\,\mathrm{OPT}_{\Pi'}(G') - \mathrm{OPT}_\Pi(G)| \leq \rho_\Pi |X'|$ and $\mathbf{tw}(G') \leq f(\tau)$; and
(R2) there is a polynomial-time algorithm that, given G and $X' \subseteq V(G)$, G', and a vertex (edge) set S' such that $\phi'(G', S')$ holds, outputs S such that $\phi(G, S)$ holds, and $||S| - |S'|| \leq \rho_\Pi |X'|$.

We put $\rho = \max\{\rho_1, \rho_\Pi\}$ and select $\epsilon' = \frac{\epsilon}{2\rho^2}$. By Lemma 7 (we are not using its full power yet), there exists γ such that, given G and X, a set X' with the following properties can be found in polynomial time

- $|X'| \leq \epsilon' |X|$, and
- for every connected component C of $G - X'$, we have that $|C \cap X| \leq \gamma$.

Since $\mathcal{G}$ is a hereditary graph class, for every connected component C of $G - X'$, we have that $C \cap X$ is a treewidth-η-modulator in graph $G[C]$. By Lemma 4 and Proposition 1, there exist $\lambda' < 1$ and β' (depending on ϵ', λ, η, and β) such that $\mathbf{tw}(G - X') \leq \beta' \gamma^{\lambda'} = \tau$. We construct G' from G and X' by making use of the polynomial-time algorithm described in (R1). Since $\mathbf{tw}(G') \leq f(\tau)$ and Π' is Min/Max-CMSO, we can use the extended version of Courcelle's theorem (Courcelle 1990, 1997) given by Borie et al. (1992) to find an optimal solution S' to Π' in $g(\epsilon')|V(G')|$ time. By the properties of the polynomial-time algorithm (R1), $||S'| - \mathrm{OPT}_{\Pi}(G)| \leq \rho|X'|$. We now call the polynomial-time algorithm described in (R2) to construct a solution S to Π from G, X', G', and S'. The conditions on the second algorithm ensure that $\phi(G, S)$ holds and that $||S| - |S'|| \leq \rho|X'|$. Hence,

$$||S| - \mathrm{OPT}_{\Pi}(G)| \leq 2\rho|X'| \leq 2\rho^2\epsilon' \,\mathrm{OPT}_{\Pi}(G) = \epsilon \,\mathrm{OPT}_{\Pi}(G).$$

Thus, for every $\epsilon > 0$, we construct an algorithm that in time $g(\epsilon)|V(G)|^{O(1)}$, where g is some function of ϵ, computes a $(1 + \epsilon)$-approximate solution to Π. This concludes the proof of the theorem. □

6.1 Approximation Schemes for Bidimensional Problems

In this section, we prove that

- every minor-bidimensional, linear-separable problem Π on a hereditary graph class $\mathcal{G}$ with the SQGM property, and
- every contraction-bidimensional, linear-separable problem Π on a hereditary graph class $\mathcal{G}$ with the SQGC property

is η-modulated. By Theorem 1, this implies the existence of EPTASs for reducible problems with these properties on the corresponding graph classes.

We need first the following lemma, whose proof is an adaptation of the proof of Fomin et al. (2010, Lemma 3.2) for our purposes. We provide the proof here for completeness.

Lemma 8. *For any $\epsilon > 0$ and minor-bidimensional, linear-separable problem Π on a hereditary graph class $\mathcal{G}$ with the SQGM property, there exists an integer $\eta \geq 0$ such that every graph $G \in \mathcal{G}$ has a treewidth-η-modulator S of size at most $\epsilon \cdot OPT_{\Pi}(G)$.*

Proof. Let β be a constant such that Π is $(\beta \cdot t)$ separable. Let α' and $0 \leq \lambda < 1$ be the constants from Lemma 1, in particular, $\mathbf{tw}(G) \leq \alpha' \cdot (OPT_{\Pi}(G))^{\lambda}$. Set $\alpha = \max\{\alpha', 1\}$. Let us note that, for any $\beta' > \beta$, Π is also $(\beta' \cdot t)$ separable; thus, we can assume that $\beta \geq 1$.

We now define a few constants. The reason that these constants are defined the way they are will become clear during the course of the proof. Finally, we set η based on α, β, λ, and ϵ.

- Set $\rho = \frac{1^{\lambda}+2^{\lambda}-3^{\lambda}}{3^{\lambda}}$ and note that $\rho > 0$.
- Set $\gamma = 4\alpha\beta$,
- set $\delta = \frac{\gamma(2\epsilon+1)}{\rho}$, and
- set $k_0 = (3 + 3\gamma)^{\frac{1}{1-\lambda}} + \frac{1}{3} \cdot (\frac{\delta}{\epsilon})^{\frac{1}{1-\lambda}}$. It is easy to verify that k_0 satisfies

$$\frac{2}{3}k_0 + \gamma k_0^{\lambda} \leq k_0 - 1 \tag{3}$$

and

$$0 \leq \frac{\epsilon \cdot k_0}{3} - \delta \left(\frac{k_0}{3}\right)^{\lambda}. \tag{4}$$

In fact, since $k_0 \geq (3 + 3\gamma)^{\frac{1}{1-\lambda}}$ and $\lambda > 0$, we have that $k_0 \geq (\frac{3}{k_0^\lambda} + 3\gamma)^{\frac{1}{1-\lambda}}$ or, equivalently, $k_0^{1-\lambda} \geq (\frac{3}{k_0^\lambda} + 3\gamma)$, which ensures that Equation (3) holds. Since $k_0 \geq \frac{1}{3} \cdot (\frac{\delta}{\epsilon})^{\frac{1}{1-\lambda}}$, we have that $k_0^{\lambda-1} \leq \frac{\delta}{\epsilon} \cdot \frac{1}{3^{\lambda-1}}$. By multiplying both sides of the inequality by $\frac{k_0^\lambda \epsilon}{3}$, we ensure the correctness of inequality (Equation (4)).

— Finally, set $\eta = \alpha \cdot k_0^\lambda$.

We prove by induction on k, that for any $k \geq \frac{1}{3}k_0$, every graph $G \in \mathcal{G}$ such that $OPT_\Pi(G) \leq k$ has a treewidth-η-modulator of size at most $\epsilon k - \delta k^\lambda$. In the base case, we consider any k such that $\frac{1}{3}k_0 \leq k \leq k_0$. By Lemma 1, any graph $G \in \mathcal{G}$ such that $OPT_\Pi(G) \leq k_0$ has treewidth at most $\alpha \cdot k_0^\lambda = \eta$. Thus, G has a treewidth-η-modulator of size 0, and

$$0 \leq \epsilon \frac{1}{3} k_0 - \delta \left(\frac{1}{3} k_0\right)^\lambda \leq \epsilon k - \delta k^\lambda$$

by the choice of k_0. In the last inequality, we used that for any $0 \leq \lambda < 1$, ϵ and δ the function $\epsilon k - \delta k^\lambda$ is monotonically increasing from the first point at which it becomes positive. This fact may easily be verified by differentiation.

For the inductive step, let $k > k_0$ and suppose that the statement is true for all values below k. We prove the statement for k. Consider a graph $G \in \mathcal{G}$ such that $OPT_\Pi(G) \leq k$. By Lemma 1, the treewidth of G is at most $\mathbf{tw}(G) \leq \alpha \cdot k^\lambda$. By Proposition 2 applied to $(G, SOL_\Pi(G))$, there is a 2/3-balanced separation (A_1, A_2) of $(G, SOL_\Pi(G))$ of order at most $\mathbf{tw}(G) + 1 \leq \alpha \cdot k^\lambda + 1$. Let $L = A_1 \setminus A_2$, $S = A_1 \cap A_2$, and $R = A_2 \setminus A_1$. Note that there are no edges from L to R. Since (A_1, A_2) is a 2/3-balanced separation, it follows that there exists a real $\frac{1}{3} \leq a \leq \frac{2}{3}$ such that $|L \cap SOL_\Pi(G)| \leq a|SOL_\Pi(G)|$ and $|R \cap SOL_\Pi(G)| \leq (1-a)|SOL_\Pi(G)|$.

Now, consider the graph $G[L \cup S]$. Since L has no neighbors in R (in G) and Π is $(\beta \cdot t)$ separable, it follows that

$$\begin{aligned} OPT_\Pi(G[L \cup S]) &\leq |SOL_\Pi(G) \cap (L \cup S)| + \beta|S| \\ &\leq ak + (\alpha k^\lambda + 1) + \beta(\alpha k^\lambda + 1) \\ &\leq ak + (\alpha k^\lambda + 1)(\beta + 1) \leq ak + \gamma k^\lambda. \end{aligned}$$

Here, the last inequality follows from the assumption that $k \geq k_0 \geq 1$ and the choice of γ. Since $k > k_0$, the properties of k_0 imply that $\frac{2}{3}k + \gamma k^\lambda \leq k - 1$. Further, $ak + \gamma k^\lambda \geq \frac{1}{3}k_0$ since $a \geq \frac{1}{3}$. Because $\mathcal{G}$ is a hereditary graph class, we may apply the induction hypothesis to $G[L \cup S]$ and obtain a treewidth-η-modulator Z_L of $G[L \cup S]$, such that

$$\begin{aligned} |Z_L| &\leq \epsilon(ak + \gamma k^\lambda) - \delta\left(ak + \gamma k^\lambda\right)^\lambda \\ &\leq \epsilon(ak + \gamma k^\lambda) - \delta k^\lambda a^\lambda. \end{aligned}$$

An identical argument applied to $G[R \cup S]$ yields a treewidth-η-modulator Z_R of $G[R \cup S]$, such that

$$|Z_R| \leq \epsilon\left((1-a)k + \gamma k^\lambda\right) - \delta k^\lambda (1-a)^\lambda.$$

We now make a treewidth-η-modulator Z of G, as follows. Let $Z = Z_L \cup S \cup Z_R$. The set Z is a treewidth-η-modulator of G because every connected component of $G - Z$ is a subset of L or R, and Z_L and Z_R are treewidth-η-modulator s for $G[L \cup S]$ and $G[R \cup S]$, respectively. Finally, we bound the size of Z.

$$\begin{aligned}|Z| &\leq |Z_L| + |Z_R| + |S| \\ &\leq \epsilon(ak + \gamma k^\lambda) - \delta k^\lambda a^\lambda + \epsilon\left((1-a)k + \gamma k^\lambda\right) - \delta k^\lambda (1-a)^\lambda + \gamma k^\lambda \\ &= \epsilon k - \delta k^\lambda \left((1-a)^\lambda + a^\lambda\right) + k^\lambda \gamma(2\epsilon + 1) \\ &\leq \epsilon k - \delta k^\lambda + k^\lambda \left(\gamma(2\epsilon+1) - \delta\rho\right) \\ &\leq \epsilon k - \delta k^\lambda.\end{aligned}$$

In the transition from the third to the fourth line, we used that $(1-a)^\lambda + a^\lambda - 1 \geq \rho$ for any a between $\frac{1}{3}$ and $\frac{2}{3}$.

To conclude the proof, we observe that the statement of the lemma follows from what has just been proved. If $OPT_\Pi(G) \leq k_0$, then G has a treewidth-η-modulator of size $\epsilon \cdot OPT_\Pi(G)$. If $OPT_\Pi(G) > k_0$, then G has a treewidth-η-modulator of size at most $\epsilon \cdot OPT_\Pi(G) - \delta(OPT_\Pi(G))^\lambda$. This completes the proof. □

Let $\mathcal{F}$ be a finite set of graphs containing at least one planar graph. It was shown in Fomin et al. (2012b) that, in this case, $\mathcal{F}$-Deletion admits a randomized constant factor approximation algorithm running in time $O(mn)$. Let us note that by the result of Robertson and Seymour (1991), every class of graphs of treewidth at most η has $f(\eta)$ minimal forbidden minors for some function f. Since any graph of treewidth η excludes an $(\eta+1)\times(\eta+1)$ grid as a minor, we have that this set of $f(\eta)$ forbidden minors contains a planar graph. Hence, Treewidth-η Modulator is a special case of $\mathcal{F}$-Deletion, and we have the following proposition.

Proposition 5 (Fomin et al. 2012b). *Treewidth-η Modulator admits a randomized $O(nm)$ time constant factor approximation algorithm.*

Lemma 8 and Proposition 5 yield the following corollary.

Lemma 9. *Let Π be a minor-(contraction-) bidimensional linear-separable problem on a graph class with the SQGM (SQGC) property. Then, there exists a constant η such that Π is η-modulated.*

Proof. Let us prove the lemma for minor-bidimensional problems; for contraction-bidimensional problems, the proof is similar.

Let Π be a minor-bidimensional, linear-separable problem on graph class $\mathcal{G}$ with the SQGM property. In order to show that Π is η-modulated, we have to show that there is a polynomial-time algorithm that, given a graph $G \in \mathcal{G}$, outputs a set X of size $O(\mathrm{OPT}_\Pi(G))$ such that $\mathbf{tw}(G-X) \leq \eta$. By Lemma 8, there is a treewidth-η-modulator of size $O(\mathrm{OPT}_\Pi(G))$. Then, by Proposition 5, a treewidth-η-modulator of size $O(\mathrm{OPT}_\Pi(G))$ can be found in polynomial time. □

Combining Theorem 1 with Lemma 9, we obtain the following result, which is the second main theorem of the article.

Theorem 2. *Let Π be a reducible*

— *minor-bidimensional, linear-separable problem and $\mathcal{G}$ a hereditary graph class with the SQGM property, or*
— *contraction-bidimensional, linear-separable problem and $\mathcal{G}$ a hereditary graph class with the SQGC property.*

Then there is an EPTAS for Π on $\mathcal{G}$.

7 MORE APPLICATIONS OF MAIN THEOREMS

By Theorem 2, all problems discussed in Section 4 admit EPTAS on graph classes with the SQGM or SQGC properties. However, there is also a range of problems that are not bidimensional, but which can be handled by Theorem 1 easily. There are also a set of problems that can be easy adapted so that either Theorem 1 or Theorem 2 can be applied.

We discuss such problems in this section.

Spanning trees and induced subgraphs. In the Max Leaf Spanning Tree problem, we are given a connected graph G and asked to find a spanning tree T of G maximizing the number of leaves of T. We could have shown that the problem is minor bidimensional and separable; however, it is easier to show that the problem is η-modulated for $\eta = 2$ directly. Kleitman and West (1991) have shown that a connected graph that contains no spanning tree with at least k leaves has at most $4k + 2$ vertices of degree at least 3. Thus, given a graph, we can just delete all vertices of degree at least 3 and the remaining graph will have treewidth at most 2. Hence, Max Leaf Spanning Tree is η-modulated.

We prove that $\Pi =$ Max Leaf Spanning Tree is reducible. The predicate $\phi_\Pi(G, S)$ defining Max Leaf Spanning Tree can be: G is connected and there is a spanning tree T of G such that each vertex of S is a leaf of T. An equivalent and, for us, a more suitable way to define $\phi_\Pi(G, S)$ is: graph $G - S$ is connected and every vertex of S has a neighbor outside S. To prove the first property of reducible problems, for a given graph G and set X, we put $G' = G - X$. Then, $\mathbf{tw}(G') \leq f(\mathbf{tw}(G - X))$ holds trivially. The intuition behind the definition of the annotated problem Π' is the following. If T is a tree in G with a set of leaves S, then $F = T - X$ is a forest. Vertices $S' = S \setminus X$ are also leaves in this forest. Moreover, every nonsingular (i.e., containing more than one vertex) connected component of F has a vertex that is not in S' and is adjacent to a vertex in X. On the other hand, every spanning forest in G' whose set of leaves contains S' and whose connected components satisfies the above property can be transformed into a spanning tree of G with at least $|S'|$ leaves by adding to F vertices of X and turning it into a tree by adding some edges not incident with vertices of S'.

Let $R = N_G(X)$. We define the annotated problem Π' with the following CMSO-expressible property $\phi_{\Pi'}$. For graph $G' = G - X$ with annotated vertex set R and vertex subset S' of G', $\phi_{\Pi'}(G', S')$ is true if

- for every vertex $v \in S'$
  either $\{v\}$ is a (singular) connected component of G' or
  v has a neighbor in $V(G') \setminus S'$, and
- for every nonsingular connected component C of G', graph $G'[C] - S'$ is connected and C contains a vertex from $R \setminus S'$.

The crucial observation is the following: for any connected graph G and every vertex subset S' of G satisfying the above conditions, we have that $G - S'$ is connected.

It is not difficult to show that Π' is a Max-CMSO problem. Now, we verify that it satisfies the first property of reducible problems. Condition $\mathbf{tw}(G') \leq f(\mathbf{tw}(G - X))$ holds trivially. To prove that for constant $\rho_\Pi = 1$, $|\,\mathrm{OPT}_{\Pi'}(G') - \mathrm{OPT}_\Pi(G)| \leq \rho_\Pi \cdot |X|$, where $\Pi =$ Max Leaf Spanning Tree, we do the following. Let S be an optimal solution to OPT_Π. Then, $S' = S \setminus X$ is a feasible solution to Π'. On the other hand, given a feasible solution S' to Π', we have that $G - S'$ is connected. Since every vertex of S' has a neighbor outside S', this implies that S' is also a feasible solution to Π. Thus, $\mathrm{OPT}_{\Pi'}(G') \leq \mathrm{OPT}_\Pi(G) \leq \mathrm{OPT}_{\Pi'}(G') + |X|$. Since a feasible solution S' of Π' is also a feasible solution to Π, we have that the second property of reducible problems also holds.

In the Max Internal Spanning Tree, we are interested to find a spanning tree with the maximum number of internal (nonleaf) vertices. It is easy to show that a maximal set of internal vertices should form a vertex cover of the input graph. Hence, the problem is η-modulated on graphs with the SQGM property. It is also easy to prove that the problem is reducible.

Now, consider the Maximum Degree Preserving Spanning Tree problem, in which, given a graph G, the objective is to find a spanning tree such that the number of vertices that have the same degree in the tree as in the input graph is maximized. Maximum Degree Preserving Spanning Tree is neither closed under taking minors nor under contractions. On the other hand, it is possible to show that every solution to the problem (Guo et al. 2010, Lemma 4.1) is a 2-dominating set in G. By Lemmata 5 and 9, r-Dominating Set is η-modulated; hence, Maximum Degree Preserving Spanning Tree is also η-modulated on graphs with the SQCM property. The proof that the problem is reducible is similar to Max Leaf Spanning Tree.

Similar arguments can be used to show that Maximum Induced Forest, Maximum Induced Bipartite Subgraph, and many other problems are η-modulated on graphs with the SQCM property. Vertices of a maximal induced forest or bipartite subgraph should form a dominating set in the input graph.

We summarize the above observations in the following lemma.

Lemma 10. *On hereditary graph classes with the SQGM property,* Max Leaf Spanning Tree *and* Max Internal Spanning Tree *are η-modulated and reducible. On hereditary graph classes with the SQCM property,* Maximum Degree Preserving Spanning Tree, Maximum Induced Forest, *and* Maximum Induced Bipartite Subgraph *are η-modulated and reducible.*

Subgraph covering and packing. Now, we consider problems about hitting and packing subgraphs. These problems can be handled in almost the same way as hitting and packing minors. Let $\mathcal{S}$ be a finite set of connected graphs. We define the following problems.

$\mathcal{S}$-Deletion
Instance: A graph G.
Objective: Find a minimum-size set $S \subseteq V(G)$ such that $G - S$ does not contain any of the graphs from $\mathcal{S}$ as a subgraph.

$\mathcal{S}$-Packing
Instance: A graph G.
Objective: Find a maximum-size collection of vertex disjoint subgraphs such that each of them contains a graph from $\mathcal{S}$ as a subgraph.

Problems $\mathcal{S}$-Deletion or $\mathcal{S}$-Packing are not bidimensional. However, we give a reduction rule that, in polynomial time, produces from a given graph G a new reduced graph G' with exactly the same values OPT for $\mathcal{S}$-Deletion and $\mathcal{S}$-Packing as in G. Moreover, the reduced graph G' has an r-dominating set of size $O(\text{OPT})$, where r is the maximum size of a graph in $\mathcal{S}$. Since r-Dominating Set is η-modulated on classes of graphs with the SGCM property, there is an algorithm that, in polynomial time, outputs a set $X \subseteq V(G)$ of size $O(\text{OPT})$ such that $\mathbf{tw}(G - X) \leq \eta$. Hence, the preprocessed versions of $\mathcal{S}$-Deletion and $\mathcal{S}$-Packing are η-modulated on graphs with the SGCM property.

We now introduce the *Redundant Vertex Rule*. Given as input G to $\mathcal{S}$-Deletion or $\mathcal{S}$-Packing, we remove all vertices that are not part of any subgraph isomorphic to any graph in $\mathcal{S}$. We can perform the Redundant Vertex Rule in $O(|V| \cdot |\mathcal{S}|)$ time by looking at a small ball around every vertex v and checking whether the ball contains a subgraph isomorphic to a graph in $\mathcal{S}$ that contains v.

This algorithm to check a subgraph isomorphic to a given graph containing a particular vertex appears in Eppstein (1999).

Consider an instance G of $\mathcal{S}$-Deletion reduced according to the Redundant Vertex Rule and let X be an optimal solution in G. Since X hits all copies of graphs in $\mathcal{S}$ occurring in G and every vertex in G appears in some copy of a graph in $\mathcal{S}$, it follows that X is an r-dominating set of G, where r is the maximum size of a graph in $\mathcal{S}$. Finally, consider an instance G of $\mathcal{S}$-Packing reduced according to the Redundant Vertex Rule, and consider an optimal solution $G_1, \ldots, G_{\mathrm{OPT}}$ such that, for every i, G_i contains a graph from $\mathcal{S}$ as a subgraph. Since every vertex of the reduced graph is in a subgraph isomorphic to a graph in $\mathcal{S}$, the selection of $G_1, \ldots, G_{\mathrm{OPT}}$ implies that every vertex v has distance at most r to some vertex in some G_i. Let $X = \{v_1, v_2, \ldots, v_{\mathrm{OPT}}\}$, where $x_i \in V(G_i)$. Then, every vertex v has distance at most $2r$ to X. Thus, Lemma 5 yields that $\mathcal{S}$-Deletion and $\mathcal{S}$-Packing are η-modulated. The proof that they are both reducible is identical to the discussion for $\mathcal{F}$-Deletion and $\mathcal{F}$-Packing.

Thus, we obtain the following lemma.

Lemma 11. *$\mathcal{S}$-Deletion or $\mathcal{S}$-Packing preprocessed with the Redundant Vertex Rule are η-modulated on graph classes with the SGCM property and are reducible.*

7.1 Partial Domination and Covering

In the Partial r-Dominating Set problem, we are given a graph G together with an integer $t \leq |V(G)|$. The objective is to find a minimum size set S such that $|B^r_G(S)| \geq t$. In Partial Vertex Cover, we are given a graph G together with an integer $t \leq |E(G)|$; the objective is to find a minimum-size vertex set S such that $|\{uv \in E(G) : u \in S \vee v \in S\}| \geq t$. We refer to edges $\{uv \in E(G) : u \in S \vee v \in S\}$ as edges *covered* by S. The PTAS for Partial Vertex Cover on planar graphs was given in Gandhi et al. (2004). To the best of our knowledge, no PTAS for Partial r-Dominating Set was known prior to our work.

We will not show that Partial r-Dominating Set and Partial Vertex Cover are bidimensional; instead, we will directly construct EPTASs for these problems on apex-minor-free graphs using the tools developed so far. We will use OPT for the size of an optimal solution to our instances. We employ an algorithm of Fomin et al. (2011b), which was developed to obtain subexponential algorithms for Partial Vertex Cover and Partial r-Dominating Set on apex-minor-free graphs. However, exactly the same algorithm works for every class of graphs $\mathcal{G}$ with the SQGC property.

Fomin et al. (2011b) provide an algorithm for solving Partial r-Dominating Set in time $2^{O(r\sqrt{\mathrm{OPT}})}n^{O(1)}$ and Partial Vertex Cover in time $2^{O(\sqrt{\mathrm{OPT}})}n^{O(1)}$. A key part of their algorithm for Partial r-Dominating Set is a polynomial-time algorithm ((Fomin et al. 2011b), Lemma 5) that, given a graph G together with integers t and k, returns an induced subgraph G' of G such that

- G' has a $3r$-dominating set of size k; and
- G has a k-sized vertex set S with $|B^r_G(S)| \geq t$ if and only if G' has a k-sized vertex set S' with $|B^r_{G'}(S')| \geq t$.

Our EPTAS loops over all possible values of k and for each such value produces G'_k from G, t, and k using Lemma 5 of Fomin et al. (2011b). If G'_k has less than t vertices, then G'_k cannot have any set that covers at least t vertices and, thus, neither can G. If G'_k has at least t vertices, we proceed with the following subroutine.

By construction, G'_k has a $3r$-dominating set of size k. We have seen already that r-Dominating Set is a contraction-bidimensional, linear-separable problem. Then, by Lemma 9, for every graph class $\mathcal{G}$ with the SQGC property, there exists a constant η such that $3r$-Dominating Set is

η-modulated in $\mathcal{G}$. This means that there is a polynomial-time algorithm that outputs a set X of size at most ρk such that $\mathbf{tw}(G'_k - X) \leq \eta$. For every $\epsilon > 0$, we define $\epsilon' = \epsilon/\rho$.

By Lemma 7 and Corollary 1, there exists a polynomial-time algorithm that computes a set X' of size at most $\epsilon'\rho k$ such that $\mathbf{tw}(G'_k - X') \leq \delta$ for a constant δ depending only on η and $\mathcal{G}$. We put all vertices of X' in our solution. Specifically, we remove X' from G'_k and put all other vertices of $B^r_{G'_k}(X')$ into a set R. Using standard dynamic programming (or by formulating the problem in an extended version of MSO (Arnborg et al. 1991)) on graphs of bounded treewidth, we can find a minimum-size set $S' \subseteq V(G'_k) \setminus X'$ such that $|X'| + |R \cup B^r_{G'_k - X'}(S')| \geq t$ in time $f(\delta)n^{O(1)}$. The subroutine returns the set $S' \cup X'$ as a solution.

Since G'_k is an induced subgraph of G, any solution $S = S' \cup X'$ returned by the subroutine covers at least t vertices in G. We return the smallest S as our approximate solution. In the iteration of the outer loop, where $k = \text{OPT}$, we have that G'_k has a set Z of size OPT that covers t vertices in G'. Observe that $Z \setminus X'$ covers at least $t - |B^r_{G'_k}(X')|$ of $V(G'_k) \setminus B^r_{G'_k}(X')$ in the graph $G'_k - X'$. Thus, the solution returned by the dynamic programming algorithm has size at most $|Z \setminus X'| \leq |Z| = \text{OPT}$ and the solution returned by the subroutine in this iteration is at most $\text{OPT} + |X'| \leq \text{OPT}(1 + \epsilon'\rho)$. Since we selected $\epsilon' = \epsilon/\rho$, we have that, for every $\epsilon > 0$, our polynomial-time algorithm returns a $(1 + \epsilon)$-approximate solution. This concludes the analysis of our EPTAS for Partial r-Dominating Set. An EPTAS for Partial Vertex Cover can be constructed in a similar manner.

Thus, we obtain the following lemma.

Lemma 12. *There is an EPTAS for* Partial r-Dominating Set *and* Partial Vertex Cover *on classes of graphs with the SQGC property.*

Let us recapitulate here the problems for which the application of Theorems 1 and 2 together with Lemmata 5, 10, 6, 11, and 12, imply EPTASs.

Corollary 2. $\mathcal{F}$-Packing, $\mathcal{F}$-Deletion, *when set* $\mathcal{F}$ *contains a planar graph,* Connected Vertex Cover, Max Internal Spanning Tree, *and* Max Leaf Spanning Tree *admit an EPTAS on hereditary graph classes with the SQGM property.*

Corollary 3. r-Dominating Set, Connected Dominating Set, r-Scattered Set, Maximum Full-Degree Spanning Tree, Maximum Induced Forest, Maximum Induced Bipartite Subgraph, Vertex-$\mathcal{S}$-Covering, Vertex-$\mathcal{S}$-Packing, Partial r-Dominating Set *and* Partial Vertex Cover *admit an EPTAS on hereditary graph classes with the SQGC property.*

8 STRUCTURE OF UNIT DISK AND MAP GRAPHS

Let $\mathcal{G}^t_U$ and $\mathcal{G}^t_M$ be the classes of unit disk and map graphs, respectively, not containing clique K_t on t vertices as a subgraph. We refer to such graphs as K_t-free graphs. In this section, we prove that both classes of graphs have SQGM property. Pipelined with Theorem 1, this implies that every η-modulated and reducible problem—in particular, each of the problems listed in Corollary 2—admits an EPTAS on $\mathcal{G}^t_U$ and $\mathcal{G}^t_M$.

8.1 K_t-free Unit Disk Graphs

We start with the following well-known observation (see Marathe et al. (1995, Lemma 3.2)) about K_t-free unit disk graphs. Recall that we use $\Delta(G)$ to denote the maximum vertex degree of graph G.

Observation 2. *For every* $G \in \mathcal{G}^t_U$, $\Delta(G) \leq 6t$.

Observation 2 allows us to prove theorems on unit disk graphs of bounded maximum degree and then use these results for K_t-free graphs.

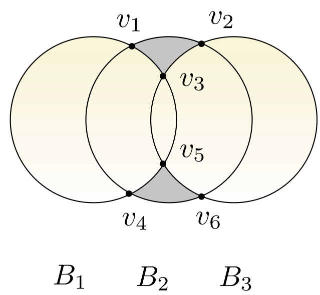

Fig. 2. A planar (multi)graph P_I formed from unit disk graph G with three vertices B_1, B_2, and B_3. The vertex set of P_i is $\{v_1, \ldots, v_6\}$. It has 7 regions. For example, for region $\mathcal{R}_1$ bounded by v_1, v_3, v_5, v_4, we have that $\mathcal{V}(\mathcal{R}_1) = \{B_1, B_2\}$. For region $\mathcal{R}_2$ bounded by v_1, v_2, v_3 and region $\mathcal{R}_3$ bounded by v_4, v_5, v_6, we have that $\mathcal{V}(\mathcal{R}_2) = \mathcal{V}(\mathcal{R}_3) = \{B_2\}$. The outer face of P_I is not a region.

Let G be a unit disk graph generated by $\mathcal{B} = \{B_1, \ldots, B_n\}$ and with $\Delta(G) = \Delta$. We associate with G an auxiliary planar graph P_G such that the treewidth of these two graphs is linearly related. Let P_I be a planar graph defined as follows. Consider the embedding (drawing) of the unit disks $\mathcal{B} = \{B_1, \ldots, B_n\}$ in the plane. Let $\mathcal{P}$ be the set of points in the plane such that each point in $\mathcal{P}$ is on the boundary of at least two disks. Essentially, this is the set of points at unit distance to centers of at least two disks. We place a vertex at each point in $\mathcal{P}$ and regard the curve between a pair of vertices as an edge. Then, the embedding of unit disks $\mathcal{B} = \{B_1, \ldots, B_n\}$ in the plane gives rise to the drawing P_I of a planar multigraph. Furthermore, let D_I be the planar dual of P_I; it is well known that D_I is also planar.

Next, we define a notion of *region* that is essential for the definition of P_G. Every face of P_I either contains no points from disks of $\mathcal{B}$ or is an intersection of interior parts of a subset of $\mathcal{B}$. We call a face $\mathcal{R}$ of the plane graph P_I a *region* if there exists a nonempty subset $\mathcal{B}' \subseteq \mathcal{B}$ of unit disks such that every point in $\mathcal{R}$ is an interior point of each disk in $\mathcal{B}'$. Hence, with every region $\mathcal{R}$, we can associate a set of unit disks. Since the vertices of G correspond to disks of $\mathcal{B}$, we can associate a subset of vertices of G—say, $\mathcal{V}(\mathcal{R})$—to a region $\mathcal{R}$. Specifically, $\mathcal{V}(\mathcal{R})$ contains all vertices in G whose disk contains $\mathcal{R}$. We remark that there could be two regions $\mathcal{R}_1$ and $\mathcal{R}_2$ with $\mathcal{V}(\mathcal{R}_1) = \mathcal{V}(\mathcal{R}_2)$. See Figure 2 for an illustration.

Now, we are ready to define the graph P_G. Let $\mathcal{R}_1, \ldots, \mathcal{R}_p$ be the regions of P_I. These are faces in P_I; hence, in the dual graph D_I, we have vertices corresponding to them. That is, in D_I for every region $\mathcal{R}_i$, we have a vertex $v(\mathcal{R}_i)$. We define

$$P_G := D_I[\{v(\mathcal{R}_i) \mid 1 \leq i \leq p\}].$$

Thus, P_G is an induced subgraph of D_I obtained by removing nonregion vertices. Figure 3 illustrates the construction of graphs P_I and P_G from unit disks drawing. Next, we prove some properties of P_G.

Lemma 13. *Let G be a unit disk graph of maximum degree Δ. Then, P_G is a planar graph and every vertex $v \in V(G)$ is a part of at most $3(\Delta^2 + \Delta)$ regions.*

Proof. The graph P_G is a subgraph of D_I, the planar dual of P_I; hence, it is also planar. Let $v \in V(G)$ be a vertex. We consider the embedding (drawing) of unit disks corresponding to the vertices of the closed neighborhood $N_G[v]$ in the plane. Then, $|N_G[v]| \leq \Delta + 1$. Let $\mathcal{L}$ be the set of the points in the plane such that each point in the set is on the boundary of at least two disks with distinct center points. This is the induced subgraph of P_I formed by the intersection points of

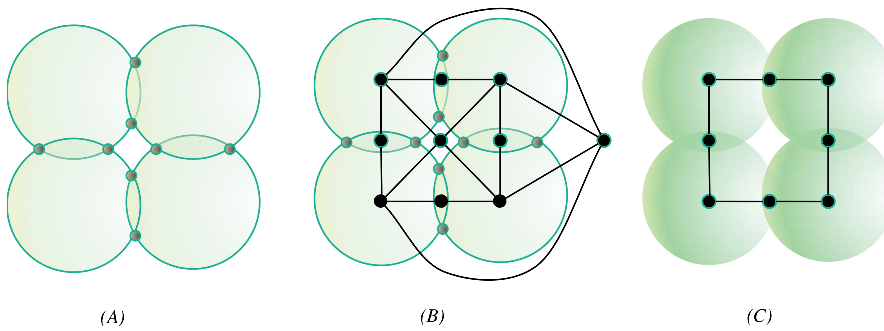

Fig. 3. (A) Drawing of a planar (multi)graph P_I formed by the drawing of four disks; (B) the dual graph D_I of P_I; (C) the graph P_G.

the boundaries of disks from $N_G[v]$. Since every two circles with distinct center points intersect in at most two points, we have that $|\mathcal{L}| \leq 2\binom{|N_G[v]|}{2} \leq 2\binom{\Delta+1}{2} = \Delta^2 + \Delta$. Consider the planar graph $P_I[\mathcal{L}]$, which is a subgraph of P_I induced by $\mathcal{L}$. Observe that v can only be a part of regions defined by faces of $P_I[\mathcal{L}]$. To obtain an upper bound on the number of faces of $P_I[\mathcal{L}]$, we first obtain an upper bound on the number of edges of $P_I[\mathcal{L}]$. First, observe that between any pair of vertices in $P_I[\mathcal{L}]$, there can be two edges at most and there are at most $|\mathcal{L}|$ pairs that have two edges between them. It is well known that a planar graph on n vertices without any parallel edges has at most $3n - 6$ edges. Thus, the number of edges in $P_I[\mathcal{L}]$ is at most $3|\mathcal{L}| - 6 + |\mathcal{L}| = 4|\mathcal{L}| - 6$. Now, by Euler's formula, the number of faces in $P_I[\mathcal{L}]$ is at most

$$2 + |E(P_I[\mathcal{L}])| - |\mathcal{L}| \leq 2 + 3|\mathcal{L}| - 6 \leq 2 + 3(\Delta^2 + \Delta) - 6 \leq 3(\Delta^2 + \Delta).$$

Thus, v is a part of at most $3(\Delta^2 + \Delta)$ regions. This completes the proof. □

Lemma 14. *Let G be a unit disk graph. Then,* $\mathbf{tw}(G) \leq (\Delta(G) + 1) \cdot (\mathbf{tw}(P_G) + 1) - 1$.

Proof. Let $\Delta(G) = \Delta$ and $(\mathcal{X}', T)$ be a tree decomposition of P_G of width $\mathbf{tw}(P_G)$. We build a tree decomposition $(\mathcal{X}, T)$ of G from the tree decomposition $(\mathcal{X}', T)$ of P_G. Let X_i' be the subset of $V(P_G)$ associated with the node i of T. We define $X_i := \bigcup_{v(\mathcal{R}) \in X_i'} \mathcal{V}(\mathcal{R})$. Recall that $\mathcal{V}(\mathcal{R})$ is a subset of vertices in $V(G)$ characterizing $\mathcal{R}$. This concludes the description of a decomposition for G. Observe that the set $\mathcal{V}(\mathcal{R})$ is contained in $N_G[w]$ for every $w \in \mathcal{V}(\mathcal{R})$; hence, the size of each of them is bounded above by $\Delta + 1$. Thus, the size of each X_i is at most $(\Delta + 1) \cdot |X_i'|$. This implies that the size of every bag X_i is at most $(\Delta + 1)(\mathbf{tw}(P_G) + 1)$.

Now, we show that this is indeed a tree decomposition for G by proving that it satisfies the three properties of a tree decomposition. By construction, every vertex of $V(G)$ is contained in some X_i. To show that for every edge $uv \in E(G)$ there is a node i such that $u, v \in X_i$, we argue as follows. If there is an edge between u and v in G, then unit disks corresponding to these vertices intersect. Hence, there is a region $\mathcal{R}$ that is completely contained inside this intersection. This implies that $u, v \in \mathcal{V}(\mathcal{R})$. For node i such that $v(\mathcal{R})$ is contained inside X_i', we have that the corresponding bag X_i contains u and v. To conclude, we need to show that, for each $v \in V(G)$, the set $Z = \{i \mid x \in X_i\}$ induces a subtree of T. Observe that v appears in all the bags corresponding to node i such that X_i' contains a vertex corresponding to a region that v is a part of. This implies that all these regions are inside the unit disk corresponding to v. Hence, the graph induced by vertices corresponding to these regions is connected. Thus, the set Z induces a subtree of T. □

We now show a linear excluded grid theorem for unit disk graphs of bounded degree. First, we need the following proposition, which is just the equivalent characterization of graph minors.

Proposition 6 (Diestel 2005). *A graph H is a minor of G if and only if there is a map $\psi : V(H) \to 2^{V(G)}$ such that, for every vertex $v \in V(H)$, $G[\psi(v)]$ is connected, for every pair of vertices $v, u \in V(H)$, $\psi(u) \cap \psi(v) = \emptyset$, and for every edge $uv \in E(H)$, there is an edge $u'v' \in E(G)$ such that $u' \in \psi(u)$ and $v' \in \psi(v)$.*

Lemma 15. *Any unit disk graph G with maximum vertex degree Δ contains a $\lfloor \frac{\mathbf{tw}(G)}{144\Delta^3} \rfloor \times \lfloor \frac{\mathbf{tw}(G)}{144\Delta^3} \rfloor$ grid as a minor.*

Proof. Let G be a unit disk graph of maximum degree Δ and define P_G as above. Since P_G is planar, by the excluded grid theorem for planar graphs (Robertson et al. 1994), P_G contains a $t \times t$ grid as a minor, where $t = \frac{\mathbf{tw}(P_G)}{6}$. By Proposition 6, we know that there is a minor model of this grid: say, $\{S[i,j] : 1 \leq i, j \leq t\}$. We know that, for every i, j, $P_G[S[i,j]]$ is connected, the sets $S[i,j]$ are pairwise disjoint, and, finally, for every i, j, i', j' such that $|i - i'| + |j - j'| = 1$, there is an edge in P_G with one endpoint in $S[i,j]$ and the other in $S[i',j']$. For every i, j, we build $S'[i,j]$ from $S[i,j]$ by replacing every vertex $v(\mathcal{R}) \in S[i,j]$ by $\mathcal{V}(\mathcal{R})$ and removing duplicates. We set $\Delta' = 3(\Delta^2 + \Delta)$, and observe that, for any vertex v in G, Lemma 13 implies that there are at most Δ' sets $S'[i,j]$ containing v.

We say that an integer pair (i,j) is *internal* if $\Delta' \leq i \leq t - \Delta'$ and $\Delta' \leq j \leq t - \Delta'$. We prove that, for any two internal pairs (i,j) and (i',j') such that $|i - i'| + |j - j'| > \Delta'$, the sets $S'[i,j]$ and $S'[i',j']$ are disjoint. To obtain a contradiction, assume that both sets contain a vertex v in G. Let X_v be the set of vertices $v(\mathcal{R})$ such that $v \in \mathcal{V}(\mathcal{R})$. We will show that $|X_v| > \Delta'$, which contradicts that v is part of at most Δ' regions. On the one hand, $P_G[X_v]$ is connected. On the other hand, both $S[i,j] \cap X_v$ and $S[i',j'] \cap X_v$ are nonempty. Since there is a unique way to draw a grid in the plane and any plane drawing of P_G must contain a drawing of the grid minor, any path in P_G between a vertex in $S[i,j]$ and a vertex in $S[i',j']$ must pass through at least $\Delta' + 1$ cycles of the grid minor. Hence, the length of a shortest path between a pair of vertices, $x \in (S[i,j] \cap X_v)$ and $y \in (S[i',j'] \cap X_v)$ in P_G, is at least $\Delta' + 1$. This implies that the length of a shortest path between x and y in $P_G[X_v]$ is at least $\Delta' + 1$; hence, $|X_v| > \Delta'$, yielding the desired contradiction. By an identical argument, one can show that, for any two internal pairs (i,j) and (i',j') such that $|i - i'| + |j - j'| > 2\Delta'$, there is no edge with one endpoint in $S'[i,j]$ and the other in $S'[i',j']$.

We can assume that $t \geq 2\Delta$. Otherwise, $\mathbf{tw}(P_G) < 12 \cdot \Delta'$. Then, by Lemma 14, $\mathbf{tw}(G) \leq (\Delta + 1) \cdot (12 \cdot \Delta' + 1)$ and the lemma trivially holds. Thus, the set of all possible pairs a, b of nonnegative integers such that $4\Delta' a + 2\Delta' \leq t$ and $4\Delta' b + 2\Delta' \leq t$ is nonempty. For all such pairs, we define the sets

— $V[a,b] = \bigcup_{i=0}^{2\Delta-1} \bigcup_{j=0}^{2\Delta-1} S[\Delta + 4\Delta a + i, \Delta + 4\Delta b + j]$.
— $E_h[a,b] = \bigcup_{i=0}^{2\Delta-1} S[3\Delta + 4\Delta a + i, 2\Delta + 4\Delta b]$.
— $E_v[a,b] = \bigcup_{j=0}^{2\Delta-1} S[2\Delta + 4\Delta a, 3\Delta + 4\Delta b + j]$.

One can think of each set $V[a,b]$ as a vertex of a grid, with each set $E_h[a,b]$ being a horizontal edge and each set $E_v[a,b]$ being a vertical edge in this grid. Build $V'[a,b]$ from $V[a,b]$ by replacing every vertex $v(\mathcal{R}) \in V[a,b]$ by $\mathcal{V}(\mathcal{R})$ and removing duplicates. Construct $E_h'[a,b]$ from $E_h[a,b]$ and $E_v'[a,b]$ from $E_v[a,b]$ similarly. We list the properties of the sets $V'[a,b]$, $E_h'[a,b]$, and $E_v'[a,b]$.

(1) For every a, b, $G[V'[a,b]]$, $G[E_h'[a,b]]$, and $G[E_v'[a,b]]$ are connected.
(2) Distinct sets $V'[a,b]$ and $V'[a',b']$ are pairwise disjoint, and there is no edge with one endpoint in $V'[a,b]$ and the other in $V'[a',b']$.
(3) For every a, b, the set $E_h'[a,b]$ is disjoint from every set $E_h'[a',b']$, $E_v'[a',b']$, and $V'[a',b']$, except possibly for $V'[a,b]$ and $V'[a+1,b]$.

(4) For every a, b, the set $E'_v[a, b]$ is disjoint from every set $E'_h[a', b']$, $E'_v[a', b']$, and $V'[a', b']$, except possibly for $V'[a, b]$ and $V'[a, b+1]$.
(5) For every a, b, there is a vertex in $E'_h[a, b]$ that is adjacent to $V'[a, b]$ and a vertex that is adjacent to $V'[a+1, b]$. Furthermore, there is a vertex in $E'_v[a, b]$ that is adjacent to $V'[a, b]$ and a vertex that is adjacent to $V'[a, b+1]$.

Property 1 follows directly from the fact that $P_G[V[a, b]]$, $P_G[E_h[a, b]]$, and $P_G[E_v[a, b]]$ are connected. Properties 2, 3, and 4 follow from the fact that, for any two internal pairs (i, j) and (i', j') such that $|i - i'| + |j - j'| > 2\Delta'$, the sets $S'[i, j]$ and $S'[i', j']$ are disjoint and have no edges between each other. Finally, Property 5 follows from the fact that, for every a, b, there is a vertex in $E_h[a, b]$ that is adjacent to $V[a, b]$ and a vertex that is adjacent to $V[a+1, b]$, and that there is a vertex in $E_v[a, b]$ that is adjacent to $V[a, b]$ and a vertex that is adjacent to $V[a, b+1]$.

For a pair a, b of integers, consider the set $E'_h[a, b]$. The properties 1, 2, and 5 ensure that some connected component $E^*_h[a, b]$ of $G[E'_h[a, b]] - (V'[a, b] \cup V'[a+1, b])$ contains at least one neighbor of $V'[a, b]$ and one neighbor of $V'[a+1, b]$. Similarly, at least one connected component $E^*_v[a, b]$ of $G[E'_v[a, b]] - (V'[a, b] \cup V'[a, b+1])$ contains at least one neighbor of $V'[a, b]$ and one neighbor of $V'[a, b+1]$. Then, the family

$$\{V'[a, b], E^*_h[a, b], E^*_v[a, b] : 4\Delta' a + 2\Delta' \leq t \text{ and } 4\Delta' b + 2\Delta' \leq t\}$$

of vertex sets in G forms a model of a $\lfloor \frac{t-2\Delta'}{4\Delta'} \rfloor \times \lfloor \frac{t-2\Delta'}{4\Delta'} \rfloor$ grid minor in G, with every edge subdivided once. We can assume that $t \geq 4\Delta'$; then, we have that $\frac{t-2\Delta'}{4\Delta'} \geq \frac{t}{8\Delta'}$.

The sets $V'[a, b]$ are models of the vertices of the grid and the sets $E^*_h[a, b]$ are models of the subdivision vertices on the horizontal edges, while $E^*_v[a, b]$ are models of the subdivision vertices on the vertical edges. Now, by Lemma 14, we know that $\mathbf{tw}(P_G) \geq \frac{\mathbf{tw}(G)+1}{(\Delta+1)} - 1$. Combining this with the fact that $t = \frac{\mathbf{tw}(P_G)}{6}$, we have that

$$\frac{t}{8\Delta'} \geq \frac{\mathbf{tw}(G)}{48\Delta' \cdot \Delta} \geq \frac{\mathbf{tw}(G)}{144\Delta^3}.$$

Thus G has a $\lfloor \frac{\mathbf{tw}(G)}{144\Delta^3} \rfloor \times \lfloor \frac{\mathbf{tw}(G)}{144\Delta^3} \rfloor$ grid as a minor. This concludes the proof. □

By Lemma 15 and Observation 2, we have the following theorem for the class $\mathcal{G}^t_U$ of unit disk graphs excluding K_t as a subgraph.

Theorem 3. *Graph class $\mathcal{G}^t_U$ has the SQGM property.*

Let us remark that, for $\mathcal{G}^t_U$, the parameter c from Definition 9 of the SQGM property is equal to 1 and the constant $\alpha = O(t^3)$.

8.2 K_t-free Map Graphs

In this section, we show that map graphs with bounded clique size have the SQGM property. As observed in Demaine et al. (2009, p. 149), for every map graph G, one can associate a map $\mathcal{M} = (\mathscr{E}, \omega)$ such that

(i) no vertex in $\mathscr{E}$ is incident only to lakes;
(ii) there are no edges in $\mathscr{E}$ whose two incident faces are both lakes (possibly the same lake);
(iii) every vertex in $\mathscr{E}$ is incident to at most one lake, and incident to such a lake at most once.

From now on, we will assume that we are given a map satisfying the above properties. For our proof, we also need the following combinatorial lemma.

Lemma 16. *Let G be a map graph associated with $\mathcal{M}$ such that the maximum clique size in G is at most t. Then, the maximum vertex degree of $\mathscr{E}$ is at most $t+2$.*

Proof. Targeting a contradiction, let us assume that there is a vertex $v \in V(\mathscr{E})$ of degree at least $t+3$. By definition, each connected component of $\mathscr{E}$ is biconnected; hence, there are at least $t+2$ cyclic faces adjacent to v. By the properties of $\mathcal{M}$, we have that all, except maybe one, adjacent faces are not lakes. However, the vertices corresponding to nation faces form a clique of size $t+1$ in G, a contradiction. □

For our proof, we also need the notions of radial and dual of map graphs. The *radial graph* $R = R(\mathcal{M})$ has a vertex for every vertex of $\mathscr{E}$ and for every nation of $\mathscr{E}$, and R is a bipartite graph with bipartition $V(\mathscr{E})$ and $N(\mathscr{E})$. Two vertices $v \in V(\mathscr{E})$ and $f \in N(\mathscr{E})$ are adjacent in R if v is incident to nation f. The *dual* $D = D(\mathcal{M})$ of $\mathcal{M}$ has vertices corresponding only to the nations of $\mathscr{E}$. The graph D has a vertex for every nation of $\mathscr{E}$, and two vertices are adjacent in D if the corresponding nations of G share an edge. We now show a linear excluded grid theorem for map graphs with bounded maximum clique.

Lemma 17. *There exists a constant ρ such that any map graph G with maximum clique size t contains a $\frac{\rho \cdot \mathbf{tw}(G)}{t} \times \frac{\rho \cdot \mathbf{tw}(G)}{t}$ grid as a minor.*

Proof. Let $\mathcal{M}$ be the map such that the graph associated with it is G. We now apply the result from Demaine et al. (2009, Lemma 4) that states that the treewidth of the map graph G is at most the product of the maximum vertex degree in $\mathscr{E}$ and $\mathbf{tw}(R)+1$. By Lemma 16, we know that the maximum vertex degree of $\mathscr{E}$ is at most $t+2$; hence, $\mathbf{tw}(G) \le (t+2)\cdot(\mathbf{tw}(R)+1)$. We now apply Demaine et al. (2009, Lemma 3), which bounds the treewidth of a radial graph of a map. In particular, by Demaine et al. (2009, Lemma 3), we have that $\mathbf{tw}(R) = O(\mathbf{tw}(D))$. This implies that $\mathbf{tw}(G) = O(t \cdot \mathbf{tw}(D))$. Observe that the graph D, the dual of $\mathcal{M}$, is a planar subgraph of G. By a result of Robertson et al. (1994), there exists a constant d such that every planar graph H contains $d \cdot \mathbf{tw}(H) \times d \cdot \mathbf{tw}(H)$ grid graph as a minor. This implies that there exists a constant d such that D has $d \cdot \mathbf{tw}(D) \times d \cdot \mathbf{tw}(D)$ grid graph as a minor. This, combined with the facts that $\mathbf{tw}(G) = O(t \cdot \mathbf{tw}(D))$ and D is a subgraph of G, implies that there exists a constant ρ such that G contains a $\frac{\rho \cdot \mathbf{tw}(G)}{t} \times \frac{\rho \cdot \mathbf{tw}(G)}{t}$ grid as a minor. □

By Lemma 15 and Observation 2, we have the following theorem about the class $\mathcal{G}_M^t$ of map graphs excluding K_t as a subgraph.

Theorem 4. *Graph class $\mathcal{G}_M^t$ has the SQGM property.*

For class $\mathcal{G}_M^t$, the parameter $c = 1$ and the constant α is of order t/ρ.

Combining Theorem 1 with Theorems 3 and 4, we arrive at the following theorem.

Theorem 5. *Let Π be a reducible minor-bidimensional problem with the separation property. There is an EPTAS for Π on $\mathcal{G}_U^t$ and $\mathcal{G}_M^t$ with runtime $O(f(\epsilon, t)\cdot n^{O(1)})$ for some function f.*

8.3 K_4-free Disk Graphs

Our result in Section 8.1 can easily be generalized to disk graphs of bounded degree. There is another widely used concept of *ply* related to geometric graphs that has turned out be very useful algorithmically (van Leeuwen 2009). An intersection graph G generated by a set of disks $\mathcal{B} = \{B_1, \ldots, B_n\}$ (not necessarily unit disks) is said to have ply ℓ if every point in the plane is contained inside at most ℓ disks in $\mathcal{B}$. Observe that if a unit disk graph has bounded ply, then it also has bounded vertex degree. This is not true for disk graphs, however. Here, we show that the classes of disk graphs with ply 3 and K_4-free disk graphs already do not have the SQGM property.

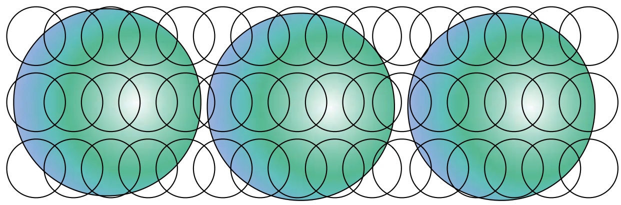

Fig. 4. Family of disks used to construct graph G_t for $t = 3$.

Theorem 6. *The classes of K_4-free disk graphs and disk graphs with ply 3 do not have the SQGM property.*

For the proof of Theorem 6, we need the concept of a bramble. A *bramble* in a graph G is a family of connected subgraphs of G such that any two of these subgraphs have a nonempty intersection or are joined by an edge. The *order* of a bramble is the minimum number of vertices required to hit all subgraphs in the bramble. Seymour and Thomas (1993) proved that a graph has treewidth k if and only if the maximum order of a bramble of G is $k + 1$. Thus, a bramble of order $k + 1$ is a witness that the graph has a treewidth at least k. We will use this characterization to get a lower bound on the treewidth of the graph that we construct.

Proof. We define a family $\mathcal{F}$ of disk graphs of ply 3 such that, for every $G \in \mathcal{F}$, we can find a set $X \subseteq V(G)$ such that $\mathbf{tw}(G - X) \leq 1$ while $\mathbf{tw}(G) \geq |X| - 1$. By Lemma 4, this would imply that $\mathcal{F}$ does not have the SQGM property.

Given a natural number $t \geq 2$, our graph G_t is defined as follows. We give the coordinates for centers of these disks.

- We have "small" disks of radius 0.99 centered at $(1.25p, 2q)$ for $0 \leq p \leq 3t^2$ and $0 \leq q \leq t - 1$.
- We have "large" disks with radius $t - 0.01$ centered at $((2p + 1)t, t)$, $0 \leq p \leq t - 1$.

Intuitively, we have small disks stacked in t rows; in each row, two consecutive disks intersect. Large disks intersect some unit disks in each row and they are pairwise disjoint among themselves (see Figure 4 for an example of our construction). Let G_t be the disk graph obtained from the intersection of disks placed as described above. Observe that every point in the plane occurs in at most 3 disks; hence, the ply of the graph is 3. Furthermore, since at most 3 disks mutually intersect, we have that G is also K_4-free. Let A be the set of vertices corresponding to small disks in rows and X be the set of remaining vertices. Observe that the graph induced by A is a set of vertex disjoint paths; hence, $\mathbf{tw}(G_t[A]) = \mathbf{tw}(G_t - X) = 1$.

We show that the treewidth of G_t is at least $t - 1$ by exhibiting a bramble of order t. Let us take the following set S_i, $0 \leq i \leq t - 1$. The set S_i consists of vertices corresponding to small disks centered at $(1.25p, 2i)$, where $0 \leq p \leq 3t^2$ and a vertex corresponding to a large disk with radius $t - 0.01$ centered at $((2i + 1)t, t)$. Let us note that graphs induced by sets S_i induce connected subgraphs of G. Since the disk with radius $t - 0.01$ intersects at least one small disk in each row, we have that, for every $i, j \in \{1, \ldots, t\}$, $i \neq j$, graphs induced by sets S_i and S_j are joined by an edge. Furthermore, $S_i \cap S_j = \emptyset$ for all $i \neq j$. This implies that the smallest number of vertices required to cover all S_i is at least t. This implies that $\mathbf{tw}(G) \geq t - 1 = |X| - 1$. □

In this section, we give approximation algorithms for several problems on unit disk graphs and map graphs. We start with the $\mathcal{F}$-Deletion problem.

Theorem 7. *Let $\mathcal{F}$ be a finite set of graphs containing a planar graph. Then, $\mathcal{F}$-Deletion admits an EPTAS on unit disk graphs and map graphs.*

Proof. Let G be the input graph, let ϵ be a fixed constant, and let $\mathcal{F}$ be an obstruction set containing a planar graph of size h. This implies that any optimal $\mathcal{F}$-deletion set in G must contain all but at most $h + 1$ vertices from any clique in G. We outline a proof below only for unit disk graphs; the proof for map graphs is similar.

The algorithm proceeds as follows. It finds a maximum clique C of G. One can find a maximum-size clique in unit disk graphs and map graphs in polynomial time (Chen et al. 1998, 2002; Clark et al. 1990; Raghavan and Spinrad 2003). The algorithm adds C to the solution and repeats this step on $G - C$ as long as there is a clique of size $\frac{(1+\epsilon)h}{\epsilon}$. Once we have that the maximum size of a clique is bounded by $\frac{(1+\epsilon)h}{\epsilon}$, we use the EPTAS obtained in Theorem 5 to get an $\mathcal{F}$-deletion set of G of size $(1 + \epsilon)$ OPT, where OPT is the size of a minimum $\mathcal{F}$-deletion set. Clearly, the set returned by the algorithm is a feasible solution. We now argue that the algorithm runs in polynomial time for every fixed ϵ.

The initial step, in which we find cliques and add all their vertices to our solution, can be done in polynomial time. Finally, we run an EPTAS on a graph in which the maximum clique size is bounded by a function of ϵ. The runtime of the algorithm guaranteed by Theorem 5 is a polynomial. Thus, the runtime has the desired form.

To bound the size of the output solution, we do the following.

Let X be an optimal $\mathcal{F}$-deletion set of G. Let $C_1, C_2, \ldots, C_t$ be the cliques found by the algorithm and G_q be the graph for which we apply Theorem 5. Since for each $i \in \{1, \ldots, t\}$, X must contain at least $|C_i| - h$ vertices of C_i and $|C_i| \geq \frac{(1+\epsilon)h}{\epsilon}$, we have that

$$|C_i| \leq (1 + \epsilon)(|C_i| - h) \leq (1 + \epsilon)(|X \cap C_i|).$$

Thus, the size of the solution returned by the algorithm satisfies the following inequality:

$$\begin{aligned}\sum_{i=1}^{t} |C_i| + (1 + \epsilon)|X \cap V(G_q)| &\leq (1 + \epsilon)\left(\sum_{i=1}^{t} |X \cap C_i| + |X \cap V(G_q)|\right)\\ &\leq (1 + \epsilon)|X| = (1 + \epsilon)\,\mathrm{OPT}.\end{aligned}$$

This completes the proof. □

Next, we show how we can obtain EPTASs for Connected Vertex Cover on unit disk graphs and map graphs.

Theorem 8. Connected Vertex Cover *admits an EPTAS on unit disk graphs and map graphs.*

Proof. For the algorithm, we need an algorithm for the annotated version of Connected Vertex Cover. Given a graph G and set X, let $G' = G - X$ and let $R = N(X)$. The annotated problem Π' with respect to set X is to find a minimum-size set $S' \subseteq V(G')$ such that every edge in G' has an endpoint in S' and every connected component of $G'[S']$ contains a vertex in R. Note that, for any connected vertex cover S of G, $S \setminus X$ is a feasible solution to Π' on G'. It is easy to show that the annotated Connected Vertex Cover is η-modulated for $\eta = 0$ and reducible. Then, by Theorem 1, Π' has EPTAS on $\mathcal{G}_U^t$ and $\mathcal{G}_M^t$.

The EPTAS for Connected Vertex Cover is very similar to the EPTAS for $\mathcal{F}$-Deletion. The only change is that we keep finding maximum cliques and including them in our solution until there is no clique of size $\frac{(2+\epsilon)}{\epsilon}$. Let $C_1, C_2, \ldots, C_q$ be the cliques found by the algorithm and G_q be the graph on which we apply Theorem 1. Let Z be the union of cliques, that is, $Z = \cup_{i \leq q} C_i$. Now, we define the annotated problem Π' with respect to set Z and using Theorem 1 obtain a set

W of size $(1+\epsilon)\,\mathrm{OPT}'$, where OPT' is the size of a minimum cardinality set in G_q such that every edge in G_q has an endpoint in W and every connected component of $G_q[W]$ contains a vertex in $R = N(Z) \cap V(G_q)$. Now, consider the set $W \cup Z$. This is a vertex cover of G and $G[W \cup Z]$ has q components. Hence, we can make $W \cup Z$ connected by adding at most $q-1$ vertices to it. Let the final solution returned by our algorithm be S. Let X be an optimal connected vertex cover of G. Since X must contain at least $|C_i| - 1$ vertices from each C_i and the size of $|C_i| \geq \frac{(2+\epsilon)}{\epsilon}$, we have that $|C_i| + 1 \leq (1+\epsilon)(|C_i| - 1) \leq (1+\epsilon)(|X \cap C_i|)$. Thus, the size of the solution returned by the algorithm satisfies the following inequality:

$$\begin{aligned}\sum_{i=1}^{q}(|C_i|+1) + (1+\epsilon)|X \cap V(G_q)| &\leq (1+\epsilon)\left(\sum_{i=1}^{q}(|X \cap C_i|) + |X \cap V(G_q)|\right)\\ &\leq (1+\epsilon)|X| = (1+\epsilon)\,\mathrm{OPT}\,.\end{aligned}$$

This completes the proof. □

PTAS for Cycle Packing on unit disk graphs. As we already have seen, Cycle Packing and, more generally, $\mathcal{F}$-Packing, are minor bidimensional, linear separable, and reducible. Thus, by Theorem 5, the problem admits an EPTAS on $\mathcal{G}_U^t$. Hence, in order to give a PTAS for Cycle Packing on unit disk graphs, it is sufficient to prove the following lemma. In particular, the following lemma implies that if we find a sufficiently large clique X, partition X into triangles, and add this partition to our packing, this will give a good approximation of how the optimum solution intersects with X. Here, a *triangle* is a cycle on three vertices.

Lemma 18. *Let G be a unit disk graph and X be a clique in G. There is a maximum-size cycle packing $C_1, C_2, \ldots, C_p$ in G such that at most 1512 cycles C_i in the packing satisfy $C_i \cap X \neq \emptyset$ and $C_i \setminus X \neq \emptyset$.*

Proof. Let X be a clique in G. The centers of all disks corresponding to vertices of X must be inside a 2×2 square. Thus, the centers of all disks corresponding to vertices in $N(X)$ must be in a 6×6 square. By Dumitrescu and Pach (2011, Lemma 2), the vertices in $N(X)$ can be partitioned into 27 cliques $S_1, S_2, \ldots, S_{27}$. Note that in the definition of unit disk graphs used in Dumitrescu and Pach (2011, Lemma 2), two vertices are adjacent if the centers of the corresponding disks are at a distance at most 1 from each other, while in this article, two vertices are adjacent if the centers of their disks are at a distance at most 2. This difference is taken into account when applying Dumitrescu and Pach (2011, Lemma 2). We say that a cycle C *crosses* X if $C \cap X \neq \emptyset$ and $C \setminus X \neq \emptyset$. Let $C_1, C_2, \ldots, C_p$ be a maximum cycle packing in G that has the fewest cycles crossing X. Observe that any cycle C that crosses X intersects with X in at most two vertices — since, otherwise, $G[C \cap X]$ induces a triangle—say, T—and then we can replace C by T in the cycle packing and obtain a maximum-size cycle packing with fewer cycles that cross X. This contradicts the choice of the packing $C_1, C_2, \ldots, C_p$.

We prove that there can be at most 54 cycles in the packing that intersect X in exactly 2 vertices. Suppose for contradiction that there are at least 55 such cycles. Each such cycle contains at least one vertex in $N(X)$. Since each vertex in $N(X)$ is in one of the 27 cliques $S_1, \ldots, S_{27}$, the pigeonhole principle implies that there are three cycles C_a, C_b, and C_c in the packing that all intersect X in exactly two vertices and a clique S_i such that $C_a \cap S_i \neq \emptyset$, $C_b \cap S_i \neq \emptyset$, and $C_c \cap S_i \neq \emptyset$. Since all cycles in the packing are vertex disjoint, this means that $S_i \cap (C_a \cup C_b \cup C_c)$ contains a triangle T_1. On the other hand, $X \cap (C_a \cup C_b \cup C_c)$ is a clique on 6 vertices and can be partitioned into two triangles T_2 and T_3. Now, we can remove C_a, C_b, and C_c from the proposed packing and replace them by T_1, T_2, and T_3. The resulting packing has the same size, but fewer cycles that cross X. This contradicts the choice of the packing $C_1, C_2, \ldots, C_p$.

Now, we show that there can be at most $2(27 \times 27) = 1458$ cycles in the packing that intersect with X in exactly 1 vertex. Every such cycle contains at least two vertices in $N(X)$. For a pair (i, j) of integers $1 \leq i \leq j \leq 27$, we say that a cycle C_a is an (i, j) cycle if C_a contains two distinct vertices u and v such that $u \in S_i$ and $v \in S_j$. If there are more than 1458 cycles in the packing that intersect with X in exactly 1 vertex, then there are i and j such that there are three (i, j)-cycles C_a, C_b, and C_c in the packing that intersect X in one vertex. Let u_a, u_b, and u_c be three vertices in $C_a \cap S_i$, $C_b \cap S_i$, and $C_c \cap S_i$ respectively. Similarly, let v_a, v_b, and v_c be the three vertices in $C_a \cap S_j$, $C_b \cap S_j$, and $C_c \cap S_j$ respectively. Now, $T_1 = \{u_a, u_b, u_c\}$, $T_2 = \{v_a, v_b, v_c\}$, and $T_3 = X \cap (C_a \cup C_b \cap C_c)$ are vertex disjoint triangles. We can replace C_a, C_b, and C_c by T_a, T_b, and T_c in the cycle packing and obtain a maximum-size cycle packing with fewer cycles that cross X, contradicting the choice of $C_1, \ldots, C_p$. Hence, there are at most $54 + 1458 = 1512$ cycles in the packing that cross X. □

Theorem 9. Cycle Packing *admits a PTAS on unit disk graphs.*

Proof. Given a unit disk graph G and ϵ, we choose t to be $\frac{(1485 \times 3) = 4455}{\epsilon}$. If G does not contain a clique of size t, then we apply the EPTAS for Cycle Packing on $\mathcal{G}_U^t$ guaranteed by Theorem 5 to give a $(1 - \epsilon)$-approximation for Cycle Packing. If G contains a clique X of size t, the algorithm partitions X into $\frac{|X|}{3}$ triangles $T_1, \ldots, T_x$, recursively finds a $(1 - \epsilon)$-approximate cycle packing $C_1, \ldots, C_p$ in $G - X$ and returns $T_1, \ldots, T_x, C_1, \ldots, C_p$ as an approximate solution. Clearly, the algorithm terminates in $n^{f(\epsilon)}$ time; thus, it remains to argue that the returned solution is indeed a $(1 - \epsilon)$-approximate cycle packing of G. We prove this by induction on the number n of vertices in G. Let OPT be the size of the largest cycle packing in G.

If there is no clique of size t and we apply the EPTAS for Cycle Packing on $\mathcal{G}_U^t$, then the returned solution is clearly a $(1 - \epsilon)$-approximation. If the algorithm finds such a clique X, Lemma 18 ensures that there is a cycle packing of size OPT such that at most 1485 cycles in the packing cross X. All cycles in this packing that intersect with X but do not cross X are triangles in X. Hence, $G - X$ contains a cycle packing of size at least $\text{OPT} - \frac{|X|}{3} - 1485$. By the inductive hypothesis, the algorithm returns a cycle packing in $G - X$ of size at least $(\text{OPT} - \frac{|X|}{3} - 1485)(1 - \epsilon)$. Now, X contains $\frac{|X|}{3}$ triangles $T_1, \ldots, T_x$. Hence, since $|X| \geq t$, the total size of the packing returned by the algorithm is at least

$$\left(\text{OPT} - \frac{|X|}{3} - 1485\right)(1 - \epsilon) + \frac{|X|}{3} = \text{OPT}(1 - \epsilon) - \left(\frac{|X|}{3} + 1485\right)(1 - \epsilon) + \frac{|X|}{3} \geq \text{OPT}(1 - \epsilon).$$

This concludes the proof. □

9 UNIT BALL GRAPHS IN $\mathbb{R}^D$: EPTAS AND SUBEXPONENTIAL ALGORITHMS

EPTAS for (Connected) Vertex Cover **on unit ball graphs in $\mathbb{R}^d$.** Recall that, by Lemma 4, the treewidth of a graph from a class with the SQGM property is sublinear in the size of a treewidth-η-modulator. In this section, we show that, for some problems we can use a weaker property, that the treewidth of every graph in a given graph class is sublinear in the number of vertices of a graph. Our results in this section are based on an observation that if for some graph class $\mathcal{G}$ the size of an optimum solution for a problem Π and the number of vertices in the input graph are linearly related, then to obtain EPTAS it is sufficient that $\mathcal{G}$ has sublinear treewidth, which is a weaker property than SQGM. The crux of this result is based on the following adaptation of the scaling lemma (Lemma 7). The proof of the following lemma is a modification of the proof of Lemma 7; we provide it here for completeness.

Lemma 19. *Let $\mathcal{G}$ be a hereditary graph class of sublinear treewidth with parameter $\lambda < 1$, that is, for every $G \in \mathcal{G}$, $\mathbf{tw}(G) = O(|V(G)|^{\lambda})$. For every $\epsilon < 1$, there is γ such that for any $G \in \mathcal{G}$, there is $X \subseteq V(G)$, satisfying*

—$|X| \leq \epsilon|V(G)|$, *and*
—*for every connected component* C *of* $G - X$, *we have that* $|C| \leq \gamma$.

Moreover, X *can be computed from* G *in polynomial time.*

Proof. Since $\mathcal{G}$ is a hereditary graph class of sublinear treewidth with parameter λ, there exists a constant β such that, for every graph $G \in \mathcal{G}$, $\mathbf{tw}(G) \leq \beta|V(G)|^\lambda$. We define $\rho = \min_{1/3 \leq \alpha \leq 2/3} \alpha^\lambda + (1-\alpha)^\lambda$. As in Lemma 7, we can assume that $\rho > 1$. We choose $\delta = \frac{\beta+1}{\rho-1}$ and $\gamma = (\frac{3\delta}{\epsilon})^{\frac{1}{1-\lambda}}$.

Let $T_\gamma(n)$ be the smallest integer such that, for every n-vertex graph $G \in \mathcal{G}$, there is $X \subseteq V(G)$ of size at most $T_\gamma(n)$ such that every connected component of $G - X$ is of size at most γ.

We claim that, for every n,

$$T_\gamma(n) \leq \epsilon n. \tag{5}$$

The proof of Equation (5) is almost identical to the proof of Equation (1) in Lemma 7. If $n \leq \gamma$, then we set $X = \emptyset$; thus, $T_\gamma(n) = 0$ and Equation (5) holds. To prove Equation (5) for larger values of n, we prove by induction on n a stronger statement if $n \geq \gamma/3$, then

$$T_\gamma(n) \leq \epsilon n - \delta n^\lambda. \tag{6}$$

We now show that if $n \geq \gamma/3$, then $T_\gamma(n) \leq \epsilon n - \delta n^\lambda$ by induction on n. For the base case if $\gamma/3 \leq n \leq \gamma$, the choice of γ implies that

$$\epsilon n - \delta n^\lambda \geq \epsilon \frac{\gamma}{3} - \delta\gamma^\lambda \geq 0.$$

As we already have shown, for $n \leq \gamma$, $T_\gamma(n) = 0$; hence, Equation (6) holds for $\gamma/3 \leq n \leq \gamma$.

We now consider $T_\gamma(n)$ for $n > \gamma$. Since the treewidth of G is at most βn^λ, we can partition $V(G)$ into L, S, and R such that $|S| \leq \beta n^\lambda + 1$, $N(L) \subseteq S$, $N(R) \subseteq S$, $|L| \leq 2n/3$, and $|R| \leq 2n/3$. Deleting S from the graph G yields two graphs $G[L]$ and $G[R]$ with no edges between them. Since $\mathcal{G}$ is a hereditary graph class, we can add S to X and then proceed recursively in $G[L]$ and $G[R]$. This yields the following recurrence for T_γ.

$$T_\gamma(n) \leq \max_{1/3 \leq \alpha \leq 2/3} T_\gamma(\alpha n) + T_\gamma((1-\alpha)n) + \beta n^\lambda + 1.$$

Observe that since $n \geq \gamma$, we have that $\alpha n \geq \gamma/3$ and $(1 - \alpha n) \geq \gamma/3$. The induction hypothesis then yields the following inequality.

$$\begin{aligned}
T_\gamma(n) &\leq \max_{1/3 \leq \alpha \leq 2/3} T_\gamma(\alpha n) + T_\gamma((1-\alpha)n) + \beta n^\lambda + 1 \\
&\leq \max_{1/3 \leq \alpha \leq 2/3} \epsilon n - \delta(\alpha n)^\lambda - \delta((1-\alpha)n)^\lambda + \beta n^\lambda + 1 \\
&\leq \max_{1/3 \leq \alpha \leq 2/3} \epsilon n - \delta n^\lambda(\alpha^\lambda + (1-\alpha)^\lambda) + \beta n^\lambda + 1 \\
&\leq \epsilon n - \delta n^\lambda - \delta(\rho - 1)n^\lambda + \beta n^\lambda + 1 \\
&\leq \epsilon n - \delta n^\lambda.
\end{aligned}$$

The last inequality holds whenever $\delta(\rho - 1)n^\lambda \geq \beta n^\lambda + 1$, which is ensured by the choice of δ and the fact that $n^\lambda \geq 1$. Thus, Equation (5) holds for all n. In other words, there exists a set X of size at most ϵn such that, for every component C of $G - X$, we have that $|C| \leq \gamma$.

To show that X can be computed from G in polynomial time, we observe that the inductive proof can be converted into a recursive algorithm. The only computationally hard step of the proof is the construction of a tree decomposition of G in each inductive step. As in Lemma 7, instead of computing the treewidth exactly, we use the $d^*\sqrt{\log \mathbf{tw}(G)}$-approximation algorithm by Feige et al.

(2008), where d^* is a fixed constant. Thus, when we partition $V(G)$ into L, S, and R, the upper bound on the size of S will be $d^*(\beta n^{\lambda} + 1)\sqrt{\log(\beta n^{\lambda})}$ instead of $\beta n^{\lambda} + 1$. However, for any $\lambda < \lambda' < 1$, there is a β' such that $d^*(\beta n^{\lambda})\sqrt{\log(\beta n^{\lambda})} < \beta' n^{\lambda'}$. Now, we can apply the above analysis with β' instead of β and λ' instead of λ to bound the size of the set X output by the algorithm. This concludes the proof of the lemma. □

Using Lemma 19, we can obtain the following analogue of Theorem 1.

Theorem 10. *Let $\mathcal{G}$ be a hereditary class of graphs with sublinear treewidth, and let Π be a reducible graph optimization problem. The* promise *version of Π, where the promise is that every instance G satisfies $\mathrm{OPT}_{\Pi}(G) = \Omega(|V(G)|)$, has an EPTAS on $\mathcal{G}$.*

Proof. For $\epsilon > 0$, we construct a polynomial time algorithm that finds a $(1 + \epsilon)$-approximation for Π. Since for every $G \in \mathcal{G}$, $\mathrm{OPT}_{\Pi}(G) = \Omega(|V(G)|)$, we have that there is a constant ρ_1 such that $\mathrm{OPT}_{\Pi}(G) \geq \rho_1 \cdot |V(G)|$. Since Π is reducible, there exists a Min/Max-CMSO problem Π', a constant ρ_{Π} and a function $f : \mathbb{N} \to \mathbb{N}$ such that

(R1) there is a polynomial-time algorithm that, given G and $X \subseteq V(G)$, outputs G' such that $|\mathrm{OPT}_{\Pi'}(G') - \mathrm{OPT}_{\Pi}(G)| \leq \rho_{\Pi}|X|$ and $\mathbf{tw}(G') \leq f(\tau)$; and
(R2) there is a polynomial-time algorithm that, given G and $X \subseteq V(G)$, G' and a vertex set S' such that $\phi_{\Pi'}(G', S')$ is true outputs S such that $\phi_{\Pi}(G, S)$ holds and $||S| - |S'|| \leq \rho_{\Pi}|X|$.

We define $\rho = \max(\rho_1, \rho_{\Pi})$ and $\epsilon' = \frac{\epsilon}{2\rho^2}$. Furthermore, since $\mathcal{G}$ is a hereditary graph class of sublinear treewidth, there are constants $\lambda < 1$ and β such that, for every $G \in \mathcal{G}$, $\mathbf{tw}(G) \leq \beta|V(G)|^{\lambda}$. By Lemma 19, there exist γ, $\lambda' < 1$ and β' depending on ϵ', λ and β such that for a given n-vertex graph G, a set $X \subseteq V(G)$ with the following properties can be found in polynomial time. First, $|X| \leq \epsilon' n$; second, for every component C of G – X, we have that $|C| \leq \gamma$. Thus, $\mathbf{tw}(G - X) = \tau \leq \beta' \gamma^{\lambda'}$.

We construct G' from G and X by making use of the polynomial-time algorithm guaranteed by (R1). Since $\mathbf{tw}(G') \leq f(\tau)$, we can use the extended version of Courcelle's theorem (Courcelle 1990, 1997) given by Borie et al. (1992) to find an optimal solution S' to Π' in $g(\epsilon')|V(G')|$ time. By (R1), $||S'| - \mathrm{OPT}_{\Pi}(G)| \leq \rho|X|$. We now use the polynomial-time algorithm guaranteed by (R2) to construct a solution S to Π from G, X, G', and S'. The properties of this algorithm ensure that $\phi_{\Pi}(G, S)$ holds and that $||S| - |S'|| \leq \rho|X|$; hence, $||S| - \mathrm{OPT}_{\Pi}(G)| \leq 2\rho|X| \leq 2\rho^2\epsilon' \, \mathrm{OPT}_{\Pi}(G)$. Since $\epsilon' = \frac{\epsilon}{2\rho^2}$, we have that $||S| - \mathrm{OPT}_{\Pi}(G)| \leq \epsilon \, \mathrm{OPT}_{\Pi}(G)$. □

Lemma 20. *Let G be an intersection graph of unit balls in $\mathbb{R}^d$, for a fixed d. If G does not contain an isolated vertex, then the minimum size of a (connected) vertex cover is at least $|V(G)|/f(d)$, where $f(d) = 2^{0.401d(1+o(1))} + 2$.*

Proof. Let G be an intersection graph of unit balls in $\mathbb{R}^d$ for a fixed d. For our proof, we need the concept of *kissing number*. The kissing number τ_d is the maximum number of nonoverlapping d-dimensional unit balls of equal size that can touch a unit ball in $\mathbb{R}^d$. It was shown in Kabatiansky and Levenshtein (1978) that $\tau_d \leq 2^{0.401d(1+o(1))}$. This implies that, for any vertex $v \in V(G)$, $N(v)$ does not contain an independent set of size bigger than $\tau_d + 1$.

Given a graph G, we compute a maximal matching, say M. Clearly, the size of M is a lower bound on the size of a minimum (connected) vertex cover. Let $V(M)$ be the set of endpoints of edges in M and $I = V(G) \setminus V(M)$. Clearly, I is an independent set. Furthermore, every vertex in I is adjacent to some vertex in $V(M)$. Hence, we have that $|I| \leq |V(M)|(\tau_d + 1)$. This implies that $|V(G)| = |V(M)| + |I| \leq 2|M| + 2|M|(\tau_d + 1)$. The last inequality implies the lemma. □

Finally, we note that every graph G—that is, an intersection graph of unit balls in $\mathbb{R}^d$, with maximum clique size Δ—has the property that every point in $\mathbb{R}^d$ is in at most Δ unit balls. This, together with the result from Miller et al. (1997), implies that the treewidth of G is $c_d \Delta^{1/d} |V(G)|^{1-\frac{1}{d}}$, where c_d is a constant depending only on d. This implies that an intersection graph of unit balls in $\mathbb{R}^d$ with bounded maximum clique has sublinear treewidth. Thus, an EPTAS for Connected Vertex Cover and Vertex Cover can be obtained along similar lines as in Theorems 7 and 8, and finally using Theorem 10 instead of Theorem 1 to arrive at the following result.

Theorem 11. Connected Vertex Cover *and* Vertex Cover *admit an EPTAS on unit ball graphs of fixed dimension.*

We can also obtain PTASs for Connected Vertex Cover and Vertex Cover on disk graphs, as a simple adaptation of the recent results by Jansen (2010) allows us to preprocess disk graphs with bounded clique size such that the size of an optimum solution and the number of vertices in the input graph are linearly related. The reason that we get PTASs rather than EPTASs is that there is no known fast algorithm for finding large cliques in a disk graph. Hence, we search for cliques of size at least $f(\epsilon)$ by brute force, yielding an $n^{f(\epsilon)}$ overhead in the runtime.

Parameterized subexponential time algorithms. In this section, we show how to obtain parameterized subexponential time algorithms for several problems. Formally, a *parameterization* of a problem assigns an integer k to each input instance and a parameterized problem is *fixed-parameter tractable* (FPT) if there is an algorithm that solves the problem in time $f(k) \cdot |I|^{O(1)}$, where $|I|$ is the size of the input and f is an arbitrary computable function. We say that a parameterized problem has a parameterized subexponential algorithm if it is solvable in time $2^{o(k)} \cdot |I|^{O(1)}$. We refer to the book (Cygan et al. 2015) for an introduction to parameterized algorithms.

Our basic idea is to find a "large" clique and guess the intersection of an optimal solution with this clique. We recursively do this until we do not have a large clique. Once we do not have large cliques, we show that the treewidth of the resulting graph must be bounded as well. At that point, we use dynamic programming on graphs of bounded treewidth to solve the problem optimally. We exemplify our approach on Feedback Vertex Set. Recall that in this problem we are given a graph G and a positive integer k, which is a parameter. The question is to check whether there is a subset $F \subseteq V(G)$, $|F| \leq k$, such that $G - F$ is acyclic. The set F is called the *feedback vertex set* of G.

Theorem 12. Feedback Vertex Set *admits a parameterized subexponential time $2^{O(k^{0.75} \log k)} n^{O(1)}$ algorithm on n-vertex unit disk graphs and map graphs.*

Proof. We give a subexponential time parameterized algorithm on map graphs. An algorithm on unit disk graphs is similar. Given k, we set the value $c = k^{\epsilon}$ for a value of ϵ to be fixed later. The algorithm will pass down the value of c to recursive calls such that c remains fixed even though k changes. The algorithm proceeds as follows. Given an instance (G, k), it finds a maximum clique C of G. Recall that we can find a maximum clique in unit disk graphs and map graphs in polynomial time (Chen et al. 1998, 2002; Clark et al. 1990; Raghavan and Spinrad 2003). If $|C| > k + 2$, then we return that G does not have feedback vertex set of size at most k. Next, we check whether $|C| \leq c$.

If $|C| \leq c$, then the considered graph is in $\mathcal{G}_M^c$. By Theorem 4, this class has the SQGM property with $c = 1$ and $\alpha = 1$. Hence, we know that $\mathbf{tw}(G) \leq O(k^{0.5+\epsilon})$. In this case, we apply the known algorithm for Feedback Vertex Set that, given a tree decomposition of width t of a graph G on n vertices, finds a minimum-size feedback vertex set in time $2^{O(t)} n^{O(1)}$ (see, e.g., Cygan et al. (2015)). Hence, in this case. the running time of our algorithm will be $2^{O(k^{0.5+\epsilon} \log k)} n^{O(1)}$.

In the case that $|C| > c$, we know that any feedback vertex set F of G contains almost all of the vertices in C, in particular, $|C \setminus F| \leq 2$. The algorithm branches on all $1 + |C| + \binom{|C|}{2}$ possibilities for $X = F \cap C$ and recursively solves the problem on $(G - X, k - |X|)$. If for some guess we have a yes answer, then we return yes; else, we return no. The runtime of this step is guided by the following recurrence $T(k) \leq \binom{|C|}{2} \cdot T(k - (|C| - 2)) + |C| \cdot T(k - (|C| - 1)) + T(k - |C|)$, where the terms $\binom{|C|}{2} \cdot T(k - (|C| - 2))$, $|C| \cdot T(k - (|C| - 1))$, $T(k - |C|)$ correspond to choosing $|C| - 2$ vertices in F from C, $|C| - 1$ vertices in F from C, and $|C|$ vertices in F from C, respectively. It follows that $T(k) \leq 3|C|^2 \cdot T(k + 2 - |C|) \leq 3|C|^2 \cdot T(k - |C|/2)$. Since $|C| \geq c$, a simple induction shows that $T(k) \leq (3c)^{4k/c}$, which again is upper bounded by $2^{O(\frac{4k \log c}{c})} \leq 2^{O(\frac{k \log k}{c})}$. Substituting k^ϵ for c yields that the total number of branches explored by the algorithm is upper bounded by $2^{O(k^{1-\epsilon} \log k)}$. Now, we choose ϵ such that the number of branches and the time spent in each branch is the same. Thus, we choose an ϵ such that $2^{O(k^{1-\epsilon} \log k)} = 2^{O(k^{0.5+\epsilon})}$. This gives us that $\epsilon = 1/4$ is asymptotically best possible. Thus, our algorithm runs in time $2^{O(k^{0.75} \log k)} n^{O(1)}$, concluding the proof. □

We remark that since the excluded grid theorem for unit disk graphs with clique size t has worse coefficient $\beta = O(t^3)$ than the coefficient $O(t)$ for map graphs, the runtime of the algorithm for Feedback Vertex Set on unit disk graphs is worse than on map graphs (but still subexponential). The subexponential parameterized algorithms for Feedback Vertex Set on unit disk graphs and map graphs extend to other problems, such as Vertex Cover and Cycle Packing. For Vertex Cover the extension is trivial, while for Cycle Packing it is not entirely obvious how to branch on a clique. If we come over a clique C, we know that by Lemma 18 there is a maximum cycle packing such that only 1512 cycles intersect C but are not contained in C. At most $3.1512 = 4536$ vertices intersect such cycles. Hence, we can guess this vertex set and for the remaining vertices of C, we pack them into triangles. It can be verified that this branching step, together with an algorithm for Cycle Packing, yields a subexponential-time parameterized algorithm for the problem.

Very recently, the superset of the authors in Fomin et al. (2017), by making use of different techniques, obtained algorithms solving Vertex Cover and Cycle Packing on unit disk graphs in time $2^{O(\sqrt{k} \log k)} n^{O(1)}$.

For Connected Vertex Cover, it is even more nontrivial to adapt the branching step. However, in the next lemma, we show that it is possible. In fact, we show that Vertex Cover and Connected Vertex Cover admit parameterized subexponential-time algorithms on Unit Ball Graphs in $\mathbb{R}^d$. The algorithm for Connected Vertex Cover also carries over to map graphs.

Theorem 13. Vertex Cover *and* Connected Vertex Cover *admit a parameterized subexponential-time $2^{o(k)} n^{O(1)}$ algorithm on n-vertex unit ball graphs of fixed dimension.*

Proof. Our algorithm for parameterized (Connected) Vertex Cover follows along the same line as for Feedback Vertex Set. We outline an algorithm for Connected Vertex Cover here. Let d be the dimension. The algorithm proceeds as follows. Given an instance (G, k), we first check whether $k \geq |V(G)|/f(d)$, where $f(d) = 2^{0.401d(1+o(1))} + 2$. By Lemma 20, we know that if $k < |V(G)|/f(d)$, then there is no connected vertex cover of size at most k and hence the answer is no. Else, we have that $|V(G)| = O(k)$. To implement our algorithm, we need a slight generalization of the problem considered. We keep a triple (G', k, X) for this problem, where G' is the current graph and the objective is to find a set $F \subseteq V(G')$ such that $|F| \leq k$, F is a vertex cover of G', and $G[X \cup F]$ is a connected vertex cover of G. Essentially, the graph G' will be obtained after branching on cliques and the set X will store the partially constructed solution so far. This allows us to check connectedness in the whole graph G.

Just as for Feedback Vertex Set, we set a parameter $c = k^\epsilon$ for $\epsilon = 1/(d + 1)$. The algorithm finds a maximum clique C of G'. If $|C| > k + 1$, then we return that G' does not have a desired set F of size at most k. Next, we check whether $|C| \leq c$. We first consider the case in which $|C| > k^\epsilon$.

We know that for any vertex cover F of G', we have that $|F \cap C| \geq |C| - 1$. Thus, we guess the intersection $Z = F \cap C$ and recursively solve the problem on $(G' - Z, k - |Z|, X \cup Z)$. If for some guess we have a yes answer, then we return yes; otherwise, we return no. The run time of this step is guided by the following recurrence $T(k) \leq |C| \cdot T(k - (|C| - 1)) + T(k - |C|)$, where the terms $|C| \cdot T(k - (|C| - 1))$, $T(k - |C|)$ correspond to choosing $|C| - 1$ vertices in F from C and $|C|$ vertices in F from C, respectively. This yields $T(k) \leq 2|C| \cdot T(k - |C| + 2) \leq T(k - |C|/2)$, which can be bounded by $(2c)^{2k/c}$ by a simple induction. This, in turn, is bounded by $2^{O(k^{1-\epsilon} \log k)}$.

In the other case, we have that $|C| \leq k^{\epsilon}$. As discussed before Theorem 11, by using the result from Miller et al. (1997), we have that the treewidth of G' is $c_d k^{\epsilon/d} |V(G)|^{1-\frac{1}{d}} = O(k^{1-(1-\epsilon)\frac{1}{d}})$, where c_d is a constant depending only on d. In this case, we apply a modification of the known algorithm for Connected Vertex Cover that, given a tree decomposition of width t of a graph G^* on n vertices, finds a minimum-size connected vertex cover in time $2^{O(t \log t)} n^{O(1)}$ (Moser 2005). To solve our problem, we do as follows. We first upper bound the number of connected components, η_X, in $G[X]$ by $k^{1-\epsilon}$. Recall that X has been constructed by branching on cliques of size at least $k^{\epsilon} + 1$; thus, from each such clique, we have at least k^{ϵ} vertices in X and vertices from one clique are in one component. Thus, $\eta_X \leq k/k^{\epsilon} = k^{1-\epsilon}$. Now, we construct a graph G^* as follows. Consider the graph $G[X \cup V(G')]$ and contract every connected component in $G[X]$ to a single vertex. Now, in the graph G^*, the objective is to find a connected vertex cover of size at most $k + \eta_X$ such that it contains all the vertices corresponding to connected components in $G[X]$. Now, the $\mathbf{tw}(G^*) \leq \mathbf{tw}(G') + \eta_X \leq O(k^{1-(1-\epsilon)\frac{1}{d}} + k^{1-\epsilon})$. Hence, in this case, the runtime of our algorithm is $2^{O((k^{1-(1-\epsilon)\frac{1}{d}} + k^{1-\epsilon}) \log k)} n^{O(1)}$.

Our choice of $\epsilon = 1/(d+1)$ implies that the runtime for branching on the clique is the same as the time required to run a dynamic programming algorithm on graphs of bounded treewidth. Thus, $2^{O(k^{1-\epsilon} \log k)} = 2^{O((k^{1-(1-\epsilon)\frac{1}{d}} + k^{1-\epsilon}) \log k)}$. Hence, our algorithm runs in time $2^{o(k)} n^{O(1)}$ for every fixed d, proving the theorem. □

Tractability borders. It is natural to ask how far our approach can be generalized and, in particular, whether many of the problems discussed so far have EPTASs and parameterized subexponential-time algorithms on unit ball graphs in higher than two dimensions. In this section, we show that one should not expect equally general results for unit ball graphs of at least three dimensions. In particular, we show that Feedback Vertex Set on unit ball graphs in $\mathbb{R}^3$ does not have an EPTAS unless P=NP and that the problem does not admit a subexponential-time parameterized algorithm under the Exponential Time Hypothesis of Impagliazzo, Paturi, and Zane (Impagliazzo et al. 2001).

Theorem 14. Feedback Vertex Set *on unit ball graphs in* $\mathbb{R}^3$ *does not admit a PTAS unless P = NP and has no subexponential-time parameterized algorithm unless the Exponential Time Hypothesis fails.*

A *unit ball* model of H in $\mathbb{R}^d$ is a map $f : V(H) \to \mathbb{R}^d$ such that u and v are adjacent iff the Euclidean distance between $f(u)$ and $f(v)$ is at most 1. In the construction it is much more convenient to work with this alternate definition of unit ball graphs rather than saying that $f(u)$ and $f(v)$ is at most 2; hence, we use this alternate definition in this section. In our constructions, no two vertices will map to the same point; thus, we will often refer to vertices in H by the points in $\mathbb{R}^d$ that they map to. For the proof of Theorem 14, we need the following lemmas. It appears that the following lemma can easily be derived from the results in Eades et al. (2000) about the three-dimensional orthogonal graph drawings. However, since we could not find this result explicitly, we give a proof here for completeness.

Lemma 21. *For any graph G on n vertices of maximum degree 6, there is a unit ball graph H on $O(n^2)$ vertices such that H is a subdivision of G. Furthermore, H and a unit ball model of H in $\mathbb{R}^3$ can be constructed from G in polynomial time.*

The proof of Lemma 21 is straightforward, but somewhat tedious.

Proof. In this construction, we envision the x-axis as being horizontal with positive direction toward the right, the z-axis being vertical with positive direction upward. The intuition behind the proof is that every vertex of G is assigned its own "fat" x-z plane. The edges of G are routed parallel to the y axis in the $y - x$ plane with $z = 0$; in each "fat" x-z plane, we ensure that the edges connect to their corresponding vertex. This local routing of edge endpoints to a vertex happens above the $y - x$ plane with $z = 0$ and does not interfere with the global routing of the edges.

For a point with integer coordinates (x, y, z) and integer ℓ, define the set $L[x, y, z]^{\ell}_{\mathrm{x}}$ to be $\{(x + x', y, z) : |x'| + |\ell - x'| = |\ell|\}$. In particular, if ℓ is positive, then $L[x, y, z]^{\ell}_{\mathrm{x}}$ contains $\{(x, y, z), (x + 1, y, z), (x + 2, y, z), \ldots, (x + \ell, y, z)\}$; if ℓ is negative, then $L[x, y, z]^{\ell}_{\mathrm{x}}$ contains $\{(x, y, z), (x - 1, y, z), (x - 2, y, z), \ldots, (x - \ell, y, z)\}$. Similarly, we define $L[x, y, z]^{\ell}_{\mathrm{y}}$ to be $\{(x, y + y', z) : |y'| + |\ell - y'| = |\ell|\}$ and $L[x, y, z]^{\ell}_{\mathrm{z}}$ to be $\{(x, y, z + z') : |z'| + |\ell - z'| = \ell\}$. Given three integers x, y, z, the graph $P[x, y, z]$ corresponds to the point set

$$\begin{aligned} P[x, y, z] = {} & L[x, y, z]^{-2}_{\mathrm{z}} \cup L[x, y, z - 2]^{2}_{\mathrm{x}} \\ & \cup L[x, y, z]^{2}_{\mathrm{x}} \\ & \cup L[x, y, z]^{2}_{\mathrm{z}} \cup L[x, y, z + 2]^{2}_{\mathrm{x}} \\ & \cup L[x, y, z]^{-2}_{\mathrm{x}} \cup L[x - 2, y, z]^{4}_{\mathrm{z}} \cup L[x - 2, y, z + 4]^{4}_{\mathrm{x}} \\ & \cup L[x, y, z]^{-2}_{\mathrm{y}} \cup L[x, y - 2, z]^{6}_{\mathrm{z}} \cup L[x, y - 2, z + 6]^{2}_{\mathrm{y}} \cup L[x, y, z + 6]^{2}_{\mathrm{x}} \\ & \cup L[x, y, z]^{2}_{\mathrm{y}} \cup L[x, y + 2, z]^{8}_{\mathrm{z}} \cup L[x, y + 2, z + 8]^{-2}_{\mathrm{y}} \cup L[x, y, z + 8]^{2}_{\mathrm{x}}. \end{aligned}$$

The set $P[x, y, z]$ corresponds to a vertex of degree 6 in $[x, y, z]$ and there are 6 paths, each starting in (x, y, z) and ending in $(x + 2, y, z - 2)$, $(x + 2, y, z)$, $(x + 2, y, z + 2)$, $(x + 2, y, z + 4)$, $(x + 2, y, z + 6)$, and $(x + 2, y, z + 8)$ respectively. The y-coordinate of any intermediate point on the paths is always between $y - 2$ and $y + 2$. Any points that are generated twice still correspond only to one single vertex.

For an integer y and six integers $x_1 < x_2 < \cdots < x_6$ such that $x_{i+1} - x_i \geq 2$, we define $\mathbf{P}[y, x_1, x_2, x_3, x_4, x_5, x_6]$ to be the point set

$$\begin{aligned} \mathbf{P}[y, x_1, x_2, x_3, x_4, x_5, x_6] = {} & P[-2, y, 12] \\ & \cup \bigcup_{i=1}^{6} L[0, y, 10 + 2(i - 1)]^{x_i}_{\mathrm{x}} \cup L[x_i, y, 10 + 2(i - 1)]^{-10-2(i-1)}_{\mathrm{z}}. \end{aligned}$$

The set $\mathbf{P}[y, x_1, x_2, x_3, x_4, x_5, x_6]$ corresponds to a vertex of degree 6 in $[-2, y, 12]$ with 6 paths starting in this vertex end ending in $[x_i, y, 0]$ for $1 \leq i \leq 6$. The y-coordinate of the intermediate vertices on the path is between $y - 2$ and $y + 2$. In this sense, the paths corresponding to the vertex in $[-2, y, 12]$ are routed in a "fat" $x - z$ plane.

We are now ready to construct H given G. We give the construction for 6-regular graphs G and then explain how to modify the construction to the case in which G has maximum degree 6. We label the vertices in G by $v_1, \ldots, v_n$ and the edges of G by $e_1, \ldots, e_m$ with $m \leq 3n$. For every $i \leq m$, define $a(i)$ and $b(i)$ such that the endpoints of the edge e_i are $v_{a(i)}$ and $v_{b(i)}$, respectively. Now, for every vertex v_i, let $x^i_1 < x^i_2 < \cdots < x^i_6$ be integers so that v_i is incident to the edges $e_{x^i_j}$ for $1 \leq j \leq 6$. For every vertex v_i, we add the point set $\mathbf{P}[10i, 2x^i_1, 2x^i_2, 2x^i_3, 2x^i_4, 2x^i_5, 2x^i_6]$. Finally, for every edge e_i, we add the set $L[2i, 10a(i), 0]^{10(b(i)-a(i))}_{\mathrm{y}}$. This concludes the construction of H.

It is easy to see that H can be constructed from G in polynomial time. Furthermore, it is easy to verify that H has $O(n^2)$ vertices since $m \leq 3n$. To see that H is a subdivision of G, observe that when G has an edge e_t between v_i and v_j, in H there is a path from the point $[-2, 10i, 12]$ through $[2t, 10i, 0]$ and $[2t, 10j, 0]$ to the point $[-2, 10j, 12]$. This concludes the proof of the lemma. □

LEMMA 22. *There is a polynomial-time algorithm that, given a graph G on n vertices of maximum degree* 3, *outputs a unit ball graph H together with a unit ball model of H in $\mathbb{R}^3$ such that, given any vertex cover C of G, a feedback vertex set S of H of size at most $|C|$ can be computed in polynomial time and, given any feedback vertex set S of H, a vertex cover C of G of size at most $|S|$ can be computed in polynomial time.*

PROOF. Given G, we start by applying the well-known construction for transforming instances of VERTEX COVER to instances of FEEDBACK VERTEX SET. We construct G' from G by adding a vertex x_{uv} for every edge uv of G and making x_{uv} adjacent to u and to v. Since the maximum degree of G was 3, the maximum degree of G' is 6. Now, we apply Lemma 21 to G' and obtain the graph H and a unit ball model of H. Every vertex cover C of G is a feedback vertex set of G' and since H is a subdivision of G', every vertex cover of G is a feedback vertex set of H. For the reverse direction, it is well known that, given a feedback vertex set S in a graph, one can find in polynomial time a feedback vertex set S' of size at most $|S|$ such that all vertices in S' have degree at least 3 (Bar-Yehuda et al. 1998). Let S' be a feedback vertex set of H such that every vertex in S' has degree at least 3 in H. Then, every vertex in S' is also a vertex in G. We claim that S' is a vertex cover of G. Let uv be an edge in G. Therefore, u, x_{uv}, v is a cycle in G' and since H is a subdivision of G', H contains a cycle going through u, x_{uv} and v where all vertices in the cycle except u and v have degree at most 2. Since S' is a feedback vertex set of H' containing no vertices of degree less than 3, S' contains either u or v. Hence, S' is a vertex cover of G. □

If a subexponential-time parameterized algorithm for FEEDBACK VERTEX SET on unit ball graphs in $\mathbb{R}^3$ existed, we could combine it with Lemma 22 to get a subexponential time algorithm for VERTEX COVER on graphs of maximum degree 3. Similarly, a PTAS for FEEDBACK VERTEX SET on unit ball graphs in $\mathbb{R}^3$ could be combined with Lemma 22 to yield a PTAS for VERTEX COVER on graphs of maximum degree 3. Since VERTEX COVER is known not to admit a $(1 + \epsilon)$-factor approximation algorithm, for some fixed $\epsilon > 0$, on graphs of degree at most 3 unless $P = NP$ (Alimonti and Kann 2000), and not to have subexponential-time parameterized algorithms on graphs of degree at most 3 under the Exponential Time Hypothesis (Impagliazzo et al. 2001), we obtain Theorem 14.

10 CONCLUDING REMARKS AND OPEN PROBLEMS

Derandomization. The EPTASs developed in this article are randomized. Essentially, we use randomness in two places. First, the treewidth approximation in Feige et al. (2008) uses an adaptation of the Arora, Rao, and Vazirani algorithm (Arora et al. 2009), which is randomized. Second, Proposition 5 provides a randomized constant-factor approximation to TREEWIDTH-η MODULATOR. We are not aware of any *deterministic* polynomial time constant-factor approximation for TREEWIDTH-η MODULATOR and leave the question about the existence of such an algorithm open. However, for several graph classes, we know how to approximate TREEWIDTH and TREEWIDTH-η MODULATOR deterministically.

On the H-minor-free graph, TREEWIDTH-η MODULATOR admits a linear kernel (Fomin et al. 2010). On this class of graphs, there is also a deterministic constant-factor approximation algorithm computing the treewidth of a graph. The kernelization algorithm uses the protrusion replacement technique developed in Bodlaender et al. (2016). Let us note that, in general, protrusion replacement does not always preserve approximation; thus, the existence of a linear kernel does not automatically yield a constant factor approximation. However, the lossless protrusion reductions developed in Fomin et al. (2012b) guarantee that for Treewidth-η Modulator the linear kernel from Fomin et al. (2010) also provides us with a constant-factor approximation for this problem on H-minor-free graphs. By pipelining with the fast deterministic "protrusion replacer" from Fomin et al. (2015), the kernelization arguments bring us to a linear time approximation algorithm. However, this approach cannot be extended to unit disk or map graphs basically because we are not able to express belonging of a graph to these classes of graphs in CMSO.

Runtimes. It should be noted that, for a fixed $\epsilon > 0$, the treewidth τ in Corollary 1 is $\mathcal{O}(1/\epsilon)$. For many problems discussed in this article, the CMSO-based algorithms on graphs of bounded treewidth could be replaced by standard dynamic programming algorithms with runtime $2^{\mathcal{O}(\mathbf{tw}(G))}n$ or $2^{\mathcal{O}(\mathbf{tw}(G)\log(\mathbf{tw}(G)))}n$ (see Cygan et al. (2015)). This leads to EPTASs with runtimes of the form $2^{\mathcal{O}(1/\epsilon)}n + n^{\mathcal{O}(1)}$ or $2^{\mathcal{O}(1/\epsilon\log(1/\epsilon))}n + n^{\mathcal{O}(1)}$.

As for the polynomial time $n^{\mathcal{O}(1)}$ component in the runtime of our algorithms, it depends on the implementation of the following steps.

- We have to approximate Treewidth-η Modulator in polynomial time, which takes randomized $O(nm)$ time. As discussed above, for H-minor-free graphs, protrusion replacement techniques imply a deterministic linear time algorithm for approximating Treewidth-η Modulator.
- In the Scaling Lemma (Lemma 7), in polynomial time, we rescale a given treewidth-η-modulator X. Here, we use the treewidth approximation algorithm from Feige et al. (2008), which is based on several high polynomial-time subroutines, including semidefinite programming. However, for specific graph classes, faster algorithms are known. For example, for planar graphs, treewidth can be approximated within a constant factor in time $\mathcal{O}(n \cdot poly(\log n))$ (Gu and Xu 2014). Plugging this into the Scaling Lemma yields that scaling of X on planar graphs can be done in time $\mathcal{O}(n \cdot poly(\log n))$.
- We construct approximate solutions by making use of reducible problems. The definition of reducible problems contains two components. First, for given graph G and set $X \subseteq V(G)$, we should be able to output graph G' such that $\mathrm{OPT}_{\Pi'}(G') = \mathrm{OPT}_{\Pi}(G) \pm \mathcal{O}(|X|)$ and $\mathbf{tw}(G') \leq f(\mathbf{tw}(G-X))$. For many natural problems, graph G' is just $G - X$ and thus can be constructed in linear time. Second, for a given graph G and $X \subseteq V(G)$, graphs G' and a vertex (edge) set $S' \subseteq V(G')$ such that $\phi_{\Pi'}(G', S')$ holds; we need to output $S \subseteq V(G)$ such that $\phi_{\Pi}(G, S) = \mathbf{true}$ and $|S| = |S'| \pm \mathcal{O}(|X|)$. Again, for many problems, the new set S can be taken as $S' \cup X$.

Therefore, for many problems on planar graphs, it is possible to replace the additive factor $n^{\mathcal{O}(1)}$ in the runtime by $\mathcal{O}(n \cdot poly(\log n))$.

We conclude with the following open question. So far, our framework implies EPTASs for contraction-bidimensional problems only on classes of apex-minor-free graphs. On the other hand, for a number of contraction-bidimensional problems, including Dominating Set, EPTASs on H-minor-free graphs are known (Dawar et al. (2006) and Grohe (2003)). An interesting question here would be to obtain a general characterization of contraction-bidimensional problems admitting EPTASs on H-minor-free graphs. Second, it is interesting to compare our framework with the framework of Dawar et al. (2006), who have shown that the restriction of a first-order logic definable optimization problem to the class of H-minor-free graphs admits EPTASs. While many η-modulated and reducible graph optimization problems like Feedback Vertex Set are not first-order definable, we do not know if the opposite is true. In other words, is the first-order logic definable graph optimization problem also η-modulated and reducible?

ACKNOWLEDGMENTS

We are grateful to Venkatesh Raman for many discussions on the topic of the article. We also thank Uriel Feige for providing explanations about the treewidth approximation.

REFERENCES

Jochen Alber, Hans L. Bodlaender, Henning Fernau, Ton Kloks, and Rolf Niedermeier. 2002. Fixed parameter algorithms for dominating set and related problems on planar graphs. *Algorithmica* 33, 4, 461–493.

Jochen Alber, Michael R. Fellows, and Rolf Niedermeier. 2004. Polynomial-time data reduction for dominating set. *Journal of the ACM* 51, 3, 363–384.

Paola Alimonti and Viggo Kann. 2000. Some APX-completeness results for cubic graphs. *Theoretical Computer Science* 237, 1–2, 123–134.

Stefan Arnborg, Jens Lagergren, and Detlef Seese. 1991. Easy problems for tree-decomposable graphs. *Journal of Algorithms* 12, 2, 308–340.

Sanjeev Arora, Satish Rao, and Umesh V. Vazirani. 2009. Expander flows, geometric embeddings and graph partitioning. *Journal of the ACM* 56, 2, 5:1–5:37.

Brenda S. Baker. 1994. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM* 41, 1, 153–180.

Reuven Bar-Yehuda, Dan Geiger, Joseph Naor, and Ron M. Roth. 1998. Approximation algorithms for the feedback vertex set problem with applications to constraint satisfaction and Bayesian inference. *SIAM Journal of Computing* 27, 4, 942–959.

Hans L. Bodlaender, Fedor V. Fomin, Daniel Lokshtanov, Eelko Penninkx, Saket Saurabh, and Dimitrios M. Thilikos. 2016. (Meta) kernelization. *Journal of the ACM* 63, 5, 44:1–44:69. http://dl.acm.org/citation.cfm?id=2973749.

Richard B. Borie, R. Gary Parker, and Craig A. Tovey. 1992. Automatic generation of linear-time algorithms from predicate calculus descriptions of problems on recursively constructed graph families. *Algorithmica* 7, 5 and 6, 555–581.

Sergio Cabello and David Gajser. 2015. Simple PTAS's for families of graphs excluding a minor. *Discrete Applied Mathematics* 189, 41–48.

Timothy M. Chan. 2003. Polynomial-time approximation schemes for packing and piercing fat objects. *Journal of Algorithms* 46, 2, 178–189. DOI: http://dx.doi.org/10.1016/S0196-6774(02)00294-8

Timothy M. Chan and Sariel Har-Peled. 2012. Approximation algorithms for maximum independent set of pseudo-disks. *Discrete & Computational Geometry* 48, 2, 373–392.

Chandra Chekuri and Julia Chuzhoy. 2016. Polynomial bounds for the grid-minor theorem. *Journal of the ACM* 63, 5, 40:1–40:65. http://dl.acm.org/citation.cfm?id=2820609.

Jianer Chen, Henning Fernau, Iyad A. Kanj, and Ge Xia. 2007. Parametric duality and kernelization: Lower bounds and upper bounds on kernel size. *SIAM Journal of Computing* 37, 4, 1077–1106.

Z.-Z. Chen. 2001. Approximation algorithms for independent sets in map graphs. *Journal of Algorithms* 41, 20–40.

Zhi-Zhong Chen, Enory Grigni, and Christos H. Papadimitriou. 1998. Planar map graphs. In *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing (STOC'98)*. ACM, 514–523.

Zhi-Zhong Chen, Enory Grigni, and Christos H. Papadimitriou. 2002. Map graphs. *Journal of the ACM* 49, 2, 127–138.

Julia Chuzhoy. 2016. Improved bounds for the excluded grid theorem. *CoRR* abs/1602.02629. (2016). http://arxiv.org/abs/1602.02629

Brent N. Clark, Charles J. Colbourn, and David S. Johnson. 1990. Unit disk graphs. *Discrete Mathematics* 86, 1–3, 165–177.

Bruno Courcelle. 1990. The monadic second-order logic of graphs I: Recognizable sets of finite graphs. *Information and Computation* 85, 12–75.

Bruno Courcelle. 1997. The expression of graph properties and graph transformations in monadic second-order logic. *Handbook of Graph Grammars and Computing by Graph Transformation*. World Scientific Publishing Co., Inc. River Edge, NJ, USA, 313–400.

Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Dániel Lokshtanov, Daniel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. 2015. *Parameterized Algorithms*. Springer, New York, NY.

Anuj Dawar, Martin Grohe, Stephan Kreutzer, and Nicole Schweikardt. 2006. Approximation schemes for first-order definable optimisation problems. *Logic in Computer Science* 411–420. DOI: http://dx.doi.org/10.1109/LICS.2006.13

Erik D. Demaine, Fedor V. Fomin, MohammadTaghi Hajiaghayi, and Dimitrios M. Thilikos. 2005b. Fixed-parameter algorithms for (k, r)-center in planar graphs and map graphs. *ACM Transactions on Algorithms* 1, 1, 33–47.

Erik D. Demaine, Fedor V. Fomin, MohammadTaghi Hajiaghayi, and Dimitrios M. Thilikos. 2005a. Subexponential parameterized algorithms on graphs of bounded genus and H-minor-free graphs. *J. ACM* 52, 6 (2005), 866–893.

Erik D. Demaine and MohammadTaghi Hajiaghayi. 2005. Bidimensionality: New connections between FPT algorithms and PTASs. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'05)*. ACM-SIAM, 590–601.

Erik D. Demaine and MohammadTaghi Hajiaghayi. 2008a. Bidimensionality. In *Encyclopedia of Algorithms*, Ming-Yang Kao (ed.). Springer, New York, NY.

Erik D. Demaine and MohammadTaghi Hajiaghayi. 2008b. The bidimensionality theory and its algorithmic applications. *The Computer Journal* 51, 3, 292–302.

Erik D. Demaine and MohammadTaghi Hajiaghayi. 2008c. Linearity of grid minors in treewidth with applications through bidimensionality. *Combinatorica* 28, 1 (2008), 19–36.

Erik D. Demaine, MohammadTaghi Hajiaghayi, and Ken-ichi Kawarabayashi. 2009. Algorithmic graph minor theory: Improved grid minor bounds and Wagner's contraction. *Algorithmica* 54, 2, 142–180.

Reinhard Diestel. 2005. *Graph Theory* (3rd ed.). Springer-Verlag, Heidelberg.

Frederic Dorn, Fedor V. Fomin, and Dimitrios M. Thilikos. 2008. Subexponential parameterized algorithms. *Computer Science Review* 2, 1, 29–39.

Adrian Dumitrescu and János Pach. 2011. Minimum clique partition in unit disk graphs. *Graphs and Combinatorics* 27, 3, 399–411. DOI: http://dx.doi.org/10.1007/s00373-011-1026-1

Peter Eades, Antonios Symvonis, and Sue Whitesides. 2000. Three-dimensional orthogonal graph drawing algorithms. *Discrete Applied Mathematics* 103, 1–3, 55–87. DOI: http://dx.doi.org/10.1016/S0166-218X(00)00172-4

David Eisenstat, Philip Klein, and Claire Mathieu. 2012. An efficient polynomial-time approximation scheme for Steiner forest in planar graphs. In *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'12).* SIAM, 626–638.

David Eppstein. 1999. Subgraph isomorphism in planar graphs and related problems. *Journal of Graph Algorithms and Applications* 3, 1–27.

David Eppstein. 2000. Diameter and treewidth in minor-closed graph families. *Algorithmica* 27, 275–291.

Thomas Erlebach, Klaus Jansen, and Eike Seidel. 2005. Polynomial-time approximation schemes for geometric intersection graphs. *SIAM Journal on Computing* 34, 6, 1302–1323. DOI: http://dx.doi.org/10.1137/S0097539702402676

Uriel Feige, MohammadTaghi Hajiaghayi, and James R. Lee. 2008. Improved approximation algorithms for minimum weight vertex separators. *SIAM Journal on Computing* 38, 2, 629–657.

Fedor V. Fomin, Petr A. Golovach, and Dimitrios M. Thilikos. 2011. Contraction obstructions for treewidth. *Journal of Combinatorial Theory Series B* 101, 5, 302–314.

Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, M. S. Ramanujan, and Saket Saurabh. 2015. Solving d-SAT via backdoors to small treewidth. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'15).* SIAM, 630–641.

Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. 2012b. Planar F-deletion: Approximation, kernelization and optimal FPT algorithms. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science (FOCS'12).* IEEE, 470–479.

Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. 2017. Finding, hitting and packing cycles in subexponential time on unit disk graphs. In *Proceedings of the 44th International Colloquium of Automata, Languages and Programming (ICALP'17) (Leibniz International Proceedings in Informatics (LIPIcs))*, Vol. 80. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 65:1–65:15.

Fedor V. Fomin, Daniel Lokshtanov, Venkatesh Raman, and Saket Saurabh. 2011a. Bidimensionality and EPTAS. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'11).* SIAM, 748–759.

Fedor V. Fomin, Daniel Lokshtanov, Venkatesh Raman, and Saket Saurabh. 2011b. Subexponential algorithms for partial cover problems. *Information Processing Letters* 111, 16, 814–818.

Fedor V. Fomin, Daniel Lokshtanov, and Saket Saurabh. 2012a. Bidimensionality and geometric graphs. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'12).* SIAM, 1563–1575.

Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Dimitrios M. Thilikos. 2010. Bidimensionality and kernels. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'10).* SIAM, 503–510.

Fedor V. Fomin and Dimitrios M. Thilikos. 2006. New upper bounds on the decomposability of planar graphs. *Journal of Graph Theory* 51, 1, 53–81.

Rajiv Gandhi, Samir Khuller, and Aravind Srinivasan. 2004. Approximation algorithms for partial covering problems. *Journal of Algorithms* 53, 1, 55–84. DOI: http://dx.doi.org/DOI: 10.1016/j.jalgor.2004.04.002

Martin Grohe. 2003. Local tree-width, excluded minors, and approximation algorithms. *Combinatorica* 23, 4, 613–632. DOI: http://dx.doi.org/10.1007/s00493-003-0037-9

Qianping Gu and Gengchun Xu. 2014. Near-linear time constant-factor approximation algorithm for branch-decomposition of planar graphs. *CoRR* abs/1407.6761. http://arxiv.org/abs/1407.6761.

Jiong Guo, Rolf Niedermeier, and Sebastian Wernicke. 2010. Fixed-parameter tractability results for full-degree spanning tree and its dual. *Networks* 56, 2, 116–130. DOI: http://dx.doi.org/10.1002/net.20353

Sariel Har-Peled and Kent Quanrud. 2015. Approximation algorithms for polynomial-expansion and low-density graphs. In *Proceedings of the 23rd Annual European Symposium on Algorithms (ESA),* Lecture Notes in Computer Science, Vol. 9294. Springer, Berlin, 717–728.

Dorit S. Hochbaum and Wolfgang Maass. 1985. Approximation schemes for covering and packing problems in image processing and VLSI. *Journal of the ACM* 32, 1, 130–136.

Harry B. Hunt, III, Madhav V. Marathe, Venkatesh Radhakrishnan, S. S. Ravi, Daniel J. Rosenkrantz, and Richard E. Stearns. 1998. NC-approximation schemes for NP- and PSPACE-hard problems for geometric graphs. *Journal of Algorithms* 26, 2, 238–274. DOI: http://dx.doi.org/10.1006/jagm.1997.0903

Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. 2001. Which problems have strongly exponential complexity. *Journal of Computer and System Sciences* 63, 4, 512–530.

Bart M. P. Jansen. 2010. Polynomial kernels for hard problems on disk graphs. In *Proceedings of the 12th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT'10), Lecture Notes in Comput. Science*, Vol. 6139. Springer, Berlin, 310–321.

G. A. Kabatiansky and V. I. Levenshtein. 1978. Bounds for packings on a sphere and in space. *Problemy Peredachi Informatsii* 14, 3–25.

Sanjeev Khanna and Rajeev Motwani. 1996. Towards a syntactic characterization of PTAS. In *STOC'96*. ACM, 329–337. DOI: http://dx.doi.org/10.1145/237814.237979

Philip N. Klein and Dániel Marx. 2014. A subexponential parameterized algorithm for Subset TSP on planar graphs. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'14)*. SIAM, 1812–1830.

Jon M. Kleinberg and Amit Kumar. 2001. Wavelength conversion in optical networks. *Journal of Algorithms* 38, 1, 25–50.

Daniel J. Kleitman and Douglas B. West. 1991. Spanning trees with many leaves. *SIAM Journal on Discrete Mathematics* 4, 1, 99–106. DOI: http://dx.doi.org/10.1137/0404010

Richard J. Lipton and Robert Endre Tarjan. 1979. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics* 36, 177–189.

Richard J. Lipton and Robert Endre Tarjan. 1980. Applications of a planar separator theorem. *SIAM Journal on Computing* 9, 615–627.

M. V. Marathe, H. Breu, H. B. Hunt, III, S. S. Ravi, and D. J. Rosenkrantz. 1995. Simple heuristics for unit disk graphs. *Networks* 25, 2, 59–68. DOI: http://dx.doi.org/10.1002/net.3230250205

Dániel Marx. 2008. Parameterized complexity and approximation algorithms. *The Computer Journal* 51, 1, 60–78.

Dániel Marx. 2013. The square root phenomenon in planar graphs. In *Proceedings of the 40th International Colloquium on Automata, Languages, and Programming (ICALP'13), Lecture Notes in Computer Science*, Vol. 7966. Springer, Berlin, 28.

Gary L. Miller, Shang-Hua Teng, William P. Thurston, and Stephen A. Vavasis. 1997. Separators for sphere-packings and nearest neighbor graphs. *Journal of the ACM* 44, 1, 1–29.

Hannes Moser. 2005. *Exact Algorithms for Generalizations of Vertex Cover*. Master's thesis. Institut für Informatik, Friedrich-Schiller-Universität, Jena, Thuringia, Germany.

Nabil H. Mustafa and Saurabh Ray. 2010. Improved results on geometric hitting set problems. *Discrete & Computational Geometry* 44, 4, 883–895.

George L. Nemhauser and Leslie E. Trotter, Jr. 1974. Properties of vertex packing and independence system polyhedra. *Mathematical Programming* 6, 48–61.

Vijay Raghavan and Jeremy Spinrad. 2003. Robust algorithms for restricted domains. *Journal of Algorithms* 48, 1, 160–172.

Neil Robertson and Paul D. Seymour. 1984. Graph minors. III. Planar tree-width. *Journal of Combinatorial Theory Series B* 36, 49–64.

Neil Robertson and Paul D. Seymour. 1991. Graph minors. X. Obstructions to tree-decomposition. *Journal of Combinatorial Theory Series B* 52, 2, 153–190.

Neil Robertson, Paul D. Seymour, and Robin Thomas. 1994. Quickly excluding a planar graph. *Journal of Combinatorial Theory Series B* 62, 323–348.

Paul D. Seymour and Robin Thomas. 1993. Graph searching and a minimax theorem for tree-width. *Journal of Combinatorial Theory Series B* 58, 239–257.

Mikkel Thorup. 1998. Map graphs in polynomial time. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science (FOCS'98)*. IEEE Computer Society, 396–405.

Erik Jan van Leeuwen. 2009. *Optimization and Approximation on Systems of Geometric Objects*. Ph.D. Dissertation. De Vries Instituut, Amsterdam, The Netherlands.

Received May 2016; revised May 2017; accepted October 2017