A Constant-Factor Approximation for Weighted Bond Cover

Eun Jung Kim
Université Paris-Dauphine, PSL University, CNRS, LAMSADE, 75016, Paris, France

Euiwoong Lee[1]
University of Michigan, Ann Arbor, MI, USA

Dimitrios M. Thilikos
LIRMM, Univ. Montpellier, CNRS, Montpellier, France

Abstract

The Weighted $\mathcal{F}$-Vertex Deletion for a class $\mathcal{F}$ of graphs asks, weighted graph G, for a minimum weight vertex set S such that $G - S \in \mathcal{F}$. The case when $\mathcal{F}$ is minor-closed and excludes some graph as a minor has received particular attention but a constant-factor approximation remained elusive for Weighted $\mathcal{F}$-Vertex Deletion. Only three cases of minor-closed $\mathcal{F}$ are known to admit constant-factor approximations, namely Vertex Cover, Feedback Vertex Set and Diamond Hitting Set. We study the problem for the class $\mathcal{F}$ of θ_c-minor-free graphs, under the equivalent setting of the Weighted c-Bond Cover problem, and present a constant-factor approximation algorithm using the primal-dual method. For this, we leverage a structure theorem implicit in [Joret et al., SIDMA'14] which states the following: any graph G containing a θ_c-minor-model either contains a large two-terminal *protrusion*, or contains a constant-size θ_c-minor-model, or a collection of pairwise disjoint *constant-sized* connected sets that can be contracted simultaneously to yield a dense graph. In the first case, we tame the graph by replacing the protrusion with a special-purpose weighted gadget. For the second and third case, we provide a weighting scheme which guarantees a local approximation ratio. Besides making an important step in the quest of (dis)proving a constant-factor approximation for Weighted $\mathcal{F}$-Vertex Deletion, our result may be useful as a template for algorithms for other minor-closed families.

2012 ACM Subject Classification Mathematics of computing → Approximation algorithms; Mathematics of computing → Combinatorial algorithms; Theory of computation → Approximation algorithms analysis; Theory of computation → Graph algorithms analysis

Keywords and phrases Constant-factor approximation algorithms, Primal-dual method, Bonds in graphs, Graph minors, Graph modification problems

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2021.7

Category APPROX

Related Version *Full Version*: https://arxiv.org/abs/2105.00857

Funding *Eun Jung Kim*: Supported by the ANR projects ASSK (ANR-18-CE40-0025-01) and ESIGMA (ANR-17-CE23-0010) from French National Research Agency.
Dimitrios M. Thilikos: Supported by the ANR projects DEMOGRAPH (ANR-16-CE40-0028), ESIGMA (ANR-17-CE23-0010), and the French-German Collaboration ANR/DFG Project UTMA (ANR-20-CE92-0027).

[1] Part of this work was done when the second author was a postdoc at New York University and supported by Simons Collaboration on Algorithms and Geometry.

© Eun Jung Kim, Euiwoong Lee, and Dimitrios M. Thilikos;
licensed under Creative Commons License CC-BY 4.0

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2021).
Editors: Mary Wootters and Laura Sanità; Article No. 7; pp. 7:1–7:14
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction and main ideas of our results

For a class $\mathcal{F}$ of graphs, the problem Weighted $\mathcal{F}$-Vertex Deletion asks, given weighted graph $G = (V, E, w)$, for a vertex set $S \subseteq V$ of minimum weight such that $G - S$ belongs to the class $\mathcal{F}$. The Weighted $\mathcal{F}$-Vertex Deletion captures classic graph problems such as Weighted Vertex Cover and Weighted Feedback Vertex Set, which corresponds to $\mathcal{F}$ being the classes of edgeless and acyclic graphs, respectively. A vast literature is devoted to the study of (Weighted) $\mathcal{F}$-Vertex Deletion for various instantiations of $\mathcal{F}$, both in approximation algorithms and in parameterized complexity. Much of the work considers a class $\mathcal{F}$ that is characterized by a set of forbidden (induced) subgraphs [41, 33, 14, 1, 28, 3, 4, 5, 6, 37] or that is minor-closed [22, 16, 18, 20, 31, 19, 30, 8, 2, 23, 21, 44, 11], thus characterized by a (finite) set of forbidden minors.

Lewis and Yannakakis [35] showed that $\mathcal{F}$-Vertex Deletion, the unweighted version of Weighted $\mathcal{F}$-Vertex Deletion, is NP-hard whenever $\mathcal{F}$ is nontrivial (there are infinitely many graphs in and outside of $\mathcal{F}$) and hereditary (is closed under taking induced subgraphs). It was also long known that $\mathcal{F}$-Vertex Deletion is APX-hard for every non-trivial hereditary class $\mathcal{F}$ [39]. So, the natural question is for which class $\mathcal{F}$, (Weighted) $\mathcal{F}$-Vertex Deletion admits constant-factor approximation algorithms.

When $\mathcal{F}$ is characterized by a finite set of forbidden induced subgraphs, a constant-factor approximation for Weighted $\mathcal{F}$-Vertex Deletion is readily derived with LP-rounding technique. Lund and Yannakakis [39] conjectured that for $\mathcal{F}$ characterized by a set of minimal forbidden induced subgraphs, the finiteness of $\mathcal{F}$ defines the borderline between approximability and inapproximability with constant ratio of $\mathcal{F}$-Vertex Deletion. This conjecture was refuted due to the existence of 2-approximation for Weighted Feedback Vertex Set [7, 12, 16]. Since then, a few more classes with an infinite set of forbidden induced subgraphs are known to allow constant-factor approximations for $\mathcal{F}$-Vertex Deletion, such as block graphs [1], 3-leaf power graphs [5], interval graphs [14], ptolemaic graphs [6], and bounded treewidth graphs [20, 23]. That is, we are only in the nascent stage when it comes to charting the landscape of (Weighted) $\mathcal{F}$-Vertex Deletion as to constant-factor approximability. In the remainder of this section, we focus on the case where $\mathcal{F}$ is a minor-closed class.

Known results on (Weighted) $\mathcal{F}$-Vertex Deletion. According to Robertson and Seymour theorem, every non-trivial minor-closed graph class $\mathcal{F}$ is characterized by a finite set, called (*minor*) *obstruction set*, of minimal forbidden minors, called (*minor*) *obstructions* [43]. It is also well-known that $\mathcal{F}$ has bounded treewidth if and only if one of the obstructions is planar [42]. Therefore, the $\mathcal{F}$-Vertex Deletion for $\mathcal{F}$ excluding at least one planar graph as a minor can be deemed a natural extension of Feedback Vertex Set. In this context, it is not surprising that $\mathcal{F}$-Vertex Deletion, for minor-closed $\mathcal{F}$, attracted particular attention in parameterized complexity, where Feedback Vertex Set was considered the flagship problem serving as an igniter and a testbed for new techniques.

For every minor-closed $\mathcal{F}$, the class of yes-instances to the decision version of $\mathcal{F}$-Vertex Deletion is minor-closed again (for every fixed size of a solution), thus there exists a finite obstruction set for the set of its yes-instances. With a minor-membership test algorithm [26], this implies that $\mathcal{F}$-Vertex Deletion is fixed-parameter tractable. The caveat is, such a fixed-parameter algorithm is non-uniform and non-constructive, and the exponential term in the running time is gigantic. Much endeavour was made to reduce the parametric dependence of such algorithms for $\mathcal{F}$-Vertex Deletion. The case when $\mathcal{F}$ has bounded treewidth is now understood well. The corresponding $\mathcal{F}$-Vertex Deletion is known to be solvable in time $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$ [20, 31] and the single-exponential dependency on k is asymptotically optimal under the *Exponential Time Hypothesis*[2] [31]. (See also [44] for recent parameterized algorithms for general minor-closed $\mathcal{F}$'s).

Turning to approximability, the (unweighted) $\mathcal{F}$-Vertex Deletion can be approximated within a constant-factor when $\mathcal{F}$ has bounded treewidth, say t, or equivalently when the obstruction set of $\mathcal{F}$ contains some planar graph. The first general result in this direction was the randomized $f(t)$-approximation of Fomin et al. [20]. Gupta et al. [23] made a further progress with an $\mathcal{O}(\log t)$-approximation algorithm. Unfortunately, such approximation algorithms whose approximation ratio depends only on $\mathcal{F}$ are not known when the input is *weighted*. A principal reason for this is that most of the techniques developed for the unweighted case do not extend to the weighted setting. In this direction, Agrawal et al. [2] presented a randomized $\mathcal{O}(\log^{1.5} n)$-approximation algorithm and a deterministic $\mathcal{O}(\log^2 n)$-approximation algorithm which run in time $n^{\mathcal{O}(t)}$ when $\mathcal{F}$ has treewidth at most t. It is reported in [2] that an $\mathcal{O}(\log n \cdot \log\log n)$-approximation can be deduced from the approximation algorithm of Bansal et al. [8] for the edge deletion variant of Weighted $\mathcal{F}$-Vertex Deletion. For the class $\mathcal{F}$ of planar graphs, Kawarabayashi and Sidiropoulos [30] presented an algorithm for $\mathcal{F}$-Vertex Deletion with polylogarithmic approximation ratio running in quasi-polynomial time. Beyond this work, no nontrivial approximation algorithm is known for $\mathcal{F}$ of unbounded treewidth.

Regarding *constant-factor* approximability for Weighted $\mathcal{F}$-Vertex Deletion with minor-closed $\mathcal{F}$, only three results are known till now. For the Weighted Vertex Cover, it was observed early that a 2-approximation can be instantly derived from the half-integrality of LP [40]. The local-ratio algorithm by Bar-Yehuda and Even [10] was presumably the first primal-dual algorithm and laid the groundwork for subsequent development of the primal-dual method.[3] For the Weighted Feedback Vertex Set, 2-approximation algorithms were proposed using the primal-dual method [7, 12, 16]. Furthermore, a constant-factor approximation algorithm was given for Weighted Diamond Hitting Set by Fiorini, Joret, and Pietropaoli [18] in 2010. To the best of our knowledge, no progress is made on approximation with constant ratio for minor-closed $\mathcal{F}$ since then.

For minor-closed $\mathcal{F}$ with graphs of bounded treewidth, the known approximation algorithms for (Weighted) $\mathcal{F}$-Vertex Deletion take one of the following two avenues. First, the algorithms in [8, 2, 23] draw on the fact that a graph of constant treewidth has a constant-size separator which breaks down the graph into *smaller* pieces. The measure for smallness is an important design feature of these algorithms. Regardless of the design specification, however, it seems there is an inherent bottleneck to extend these algorithmic strategy to handle weights while achieving a constant approximation ratio; the above results either use an algorithm for the Balanced Separator problem that does not admit a constant-factor approximation ratio, under the Small Set Expansion Hypothesis [38], or use a relationship between the size of the separator and the size of resulting pieces that do not hold for weighted graphs.

The second direction is the primal-dual method [10, 7, 12, 16, 18]. The constant-factor approximation of [20] for $\mathcal{F}$-Vertex Deletion is also based on the same core observation of the primal-dual algorithm such as [12]. The 2-approximation for Weighted Feedback

[2] The ETH states that 3-SAT on n variables cannot be solved in time $2^{o(n)}$, see [27] for more details.
[3] In this paper, we consider local-ratio and primal-dual as the same algorithms design paradigm and use the word primal-dual throughout the paper even when the underlying LP is not explicitly given. We refer the reader to the classic survey of Bar-Yehuda et al. [9] for the equivalence.

Vertex Set became available by introducing a new LP formulation which translates the property "$G - X$ is a forest" in terms of the *sum of degree contribution* of X. The idea of expressing the sparsity condition of $G - X$ in terms of the degree contribution of X again played the key role in [18] for Weighted Diamond Hitting Set. However, the (extended) sparsity inequality of [18] is highly intricate as the LP constraint describes the precise structure of diamond-minor-free graphs (after *taming* the graph via some special protrusion replacer). Therefore, expressing the sparsity condition for other classes $\mathcal{F}$ with tailor-made LP constraints is likely to be prohibitively convoluted. This implies that a radical simplification of the known algorithm for, say, Weighted Diamond Hitting Set will be necessary if one intends to apply the primal-dual method for broader classes.

Our result and the key ideas. Let θ_c be the graph on two vertices joined by c parallel edges. The central problem we study is the Weighted $\mathcal{F}$-Vertex Deletion where $\mathcal{F}$ is the class of θ_c-minor-free graphs: a weighted graph $G = (V, E, w)$ is given as input, and the goal is to find a vertex set S of minimum weight such that $G - S$ is θ_c-minor-free. We call this particular problem the Weighted c-Bond Cover problem, as we believe that this nomenclature is more adequate for reasons to be clear in Section 2. Our main result is the following.

▶ **Theorem 1.** *There is a constant-factor approximation algorithm for Weighted c-Bond Cover running in uniformly[4] polynomial time.*

Let us briefly recall the classic 2-approximation algorithms for Weighted Feedback Vertex Set [7, 12]. These algorithms repeatedly delineate a vertex subset S on which the induced subgraph contains an obstruction (a cycle), and "peel off" a weighted graph on S from the current weighted graph so that the weight of at least one vertex of the current graph drops to zero. The crux of this approach is to create a weighted graph to peel off (or *design a weighting scheme*) on which *every* (minimal) feasible solution is consistently an α-approximate solution. We remark that peeling-off of a weighted graph on S can be viewed as increasing the dual variable (from zero) corresponding to S until some dual constraint becomes tight, as articulated in [16].

If one aims to capitalize on the power of the primal-dual method for other minor-closed classes and ultimately for arbitrary $\mathcal{F}$ with graphs of bounded treewidth, more sophisticated weighting scheme is needed. As we already mentioned, this was successfully done by Fiorini, Joret and Pietropaoli [18] for Weighted Diamond Hitting Set, where their primal-dual algorithm is based on an intricate LP formulation. Our primal-dual algorithm diverges from such tactics, and instead use the next structural theorem as a guide for the weighting scheme. Before we present it, we need to define some basic concepts.

Given two disjoint subsets X, Y of $V(G)$, the *edges crossing* X and Y is the set of edges with one endpoint in X and the other in Y. Notice that θ_c is a minor of G iff G contains two disjoint connected sets X and Y crossed by c edges of G. We call the union $M := X \cup Y$ θ_c*-model* in G.

Given a positive integer c, a c*-outgrowth* of a graph G is a triple $\mathbf{K} = (K, u, v)$ where u, v are distinct vertices of G, K is a component of $G \setminus \{u, v\}$, $N_G(V(K)) = \{u, v\}$, and the graph, denoted by $K^{(x,y)}$, obtained from $G[V(K) \cup \{u, v\}]$ if we remove all edges with endpoints u and v is θ_c-minor free. The *size* of a c-outgrowth of G is the size of K. A *cluster collection*

[4] We use the term "unformly polynomial" in order to indicate that a constructive algorithm exists that, for every c, runs in $f(c) \cdot n^{\mathcal{O}(1)}$ time for some constructible function f.

of a graph G is a non-empty collection $\mathcal{C} = \{C_1, \ldots, C_r\}$ of pairwise disjoint non-empty connected subsets of $V(G)$. In case $\bigcup_{C\in\mathcal{C}} C = V(G)$ we say that $\mathcal{C}$ is a *cluster partition* of G. The *capacity* of a cluster collection $\mathcal{C}$ is the maximum number of vertices of a cluster in $\mathcal{C}$. We use the notation $G/\mathcal{C}$ for the multigraph obtained from $G[\bigcup_{C\in\mathcal{C}} C]$ by contracting[5] all edges in $G[C_i]$ for each $i \in \{1, \ldots, r\}$.

▶ **Theorem 2.** *There is a function $f_1 : \mathbb{N}^2 \to \mathbb{N}$ such that, for every two positive integers c and t, there is a uniformly polynomial time algorithm that, given as input a graph G, outputs one of the following:*
1. *a c-outgrowth of size at least c, or*
2. *a θ_c-model M of G of size at most $f_1(c, t)$, or*
3. *a cluster collection $\mathcal{C}$ of G of capacity at most $f_1(c, t)$ such that $\delta(G/\mathcal{C}) \geq t$, or*
4. *a report that G is θ_c-minor free.*

(By $\delta(G)$ we denote the minimum *edge-degree* of a vertex in G. The *edge-degree* of a vertex v of G, denoted by $\mathsf{edeg}_G(v)$, is the number of edges that are incident to v.) A variant of Theorem 2 was originally proved by Joret et al. [29] without the capacity condition on a cluster collection in Case 3. It turns out that imposing the capacity condition of Case 3 is crucial for designing a weighting scheme.

At each iteration, our primal-dual algorithm invokes Theorem 2. Depending on the outcome, the algorithm either runs a *replacer* (defined in Section 2) and reduces the size of a c-outgrowth, or computes a suitable weighted graph which we call *α-thin layer* (defined in Section 3), using a suitable weighting scheme, thus reducing the current weight. In both cases, we convert the current weighted graph $G = (V, E, w)$ into a new weighted graph $G' = (V', E', w')$ on a strictly smaller number of vertices so that an α-approximate solution for G' implies an α-approximate solution for G for some particular value of α.

We stress that the replacer is compatible with any approximation ratio in the sense that the optimal weight of a solution is unchanged and every solution after the replacement can be transformed to a solution that is at least as good. When Theorem 2 reports a constant-sized θ_c-model, it is easy to see that a uniformly weighted α-thin layer suffices. The gist of Theorem 2 is in the third case, which promises a collection of pairwise disjoint *constant-sized* connected sets.

Let us first consider the simplest such case where all connected sets are singletons, namely when $\delta(G) \geq t$. It is not difficult to see that, if we consider $t := 6c$ and under the *edge-degree-proportional* weight function, that is for every $v \in V(G)$, $w(v) := \mathsf{edeg}_G(v)$, any feasible solution to WEIGHTED c-BOND COVER is a 4-approximate solution.

In the general case where we have a collection of pairwise disjoint connected sets, each of size at most r, the critical observation (Lemma 3) is that if the contraction of these sets yields a graph of minimum edge-degree at least $t := 8c$, then a weighting scheme akin to the simple case also works. That is, any feasible solution to WEIGHTED c-BOND COVER is a $4r$-approximate solution. The overall primal-dual framework is summarized in Section 3.

2 Preliminary definitions and results

We use $\mathbb{N}$ for the set of non-negative integers and $\mathbb{R}_{\geq 0}$ for the set of non-negative reals. Given some $r \in \mathbb{N}$, we define $[r] = \{1, \ldots, r\}$. Given some collection $\mathcal{A}$ of objects on which the union operation can be defined, we define $\bigcup\mathcal{A} = \bigcup_{A\in\mathcal{A}} A$. All graphs we consider are multigraphs

[5] When considering edge contractions we sum up edge multiplicities of multiple edges that are created during the contraction. However, when a loop appears after a contraction, then we suppress it.

without loops. We denote a graph by $G = (V, E)$ where V and E are its vertex and edge set respectively. A vertex-weighted graph is denoted by $G = (V, E, w)$ where $w : V(G) \to \mathbb{R}_{\geq 0}$ and we say that G is a w-weighted graph. We use $V(G)$ and $E(G)$ for the vertex set and the edge multiset of G. We also refer to $|V(G)|$ as the *size* of G. If $X \subseteq V(G)$, we denote by $G[X]$ the subgraph of G induced by X and by $G - X$ the graph $G[V(G) \setminus X]$. We say that X is *connected in* G if $G[X]$ is connected. A graph H is a *minor* of a graph G if H can be obtained from a subgraph of G after contracting edges. Given a graph H, we say that G is H*-minor free* if G does not contain H as a minor. We denote by $N_G(v)$ the set of all neighbors of v in G.

Covering bonds. Let G be a graph. Given a bipartition $\{V_1, V_2\}$ of $V(G)$, the set of edges crossing V_1 and V_2 is called the *cut* of $\{V_1, V_2\}$ and an edge set is a *cut* if it is a cut of some vertex bipartition. A minimal non-empty cut is known as a *bond* in the literature. We remark that the bonds of G are precisely the circuits of the cographic matroid of G. Given a positive integer c, a c*-bond* of a graph G is any minimal cut of G of size at least c. The problem of finding the maximum c for which a graph G contains a c-bond has been examined both from the approximation [25, 15] and the parameterized point of view [17]. Given a set $S \subseteq V(G)$, we say that S is a c*-bond cover* of G if $G - S$ is θ_c-minor free. Notice that S is a c-bond cover iff $G \setminus S$ does not contain a c-bond. Given a weighted graph $G = (V, E, w)$ with $w : V(G) \to \mathbb{R}_{\geq 0}$, a *minimum weight* c*-bond cover* of G is a c-bond cover S where the weight of S, defined as $w(S) := \sum_{v \in S} w(v)$, is minimized.

It is easy to prove that, for every $c \in \mathbb{N}$, a graph G contains θ_c as a minor iff it has a c-bond. This means that when $\mathcal{F}$ is the class of θ_c-minor-free graphs, then Weighted $\mathcal{F}$-Vertex Deletion can be restated as follows.

Weighted c-Bond Cover
Input: a vertex weighted graph $G = (V, E, w)$.
Solution: a minimum weight c-bond cover of G.

The weighting scheme. Let G be a graph and let $\mathcal{C}$ be a cluster partition of G of capacity at most r. Given a cluster $C \in \mathcal{C}$ we denote by $\mathsf{ext}_{\mathcal{C}}(C)$ (or simply $\mathsf{ext}(C)$) the set of edges with one endpoint in C and the other not in C.

Let now G be an instance of Weighted c-Bond Cover for some positive integer c. We define the vertex weighting function $w_{\mathcal{C}} : V(G) \to \mathbb{R}_{\geq 0}$ so that if $v \in C \in \mathcal{C}$, then

$$w_{\mathcal{C}}(v) \quad = \quad \frac{|\mathsf{ext}(C)|}{|C|}. \tag{1}$$

When $\mathcal{C}$ is clear from the context, we simply write w instead $w_{\mathcal{C}}$. The main result of this section is that, with respect to the weight function w in Equation 1, every c-bond cover of G is a $4r$-approximation.

▶ **Lemma 3.** *Let c be a non negative integer, G be a graph, r be a positive integer, $\mathcal{C}$ be a cluster partition of G of capacity at most r and such that $\delta(G/\mathcal{C}) \geq 8c$, and $w : V(G) \to \mathbb{R}_{\geq 0}$ be a vertex weighting function as in* Equation 1*. Then for every c-bond cover X of G, it holds that $\frac{1}{2r} \cdot |E(G/\mathcal{C})| \leq \sum_{v \in X} w(v) \leq 2 \cdot |E(G/\mathcal{C})|$.*

Proof. For the upper bound, note that $\sum_{v \in X} w(v) = \sum_{v \in X} \frac{|\mathsf{ext}(C)|}{|C|} \leq \sum_{C \in \mathcal{C}} \sum_{v \in C} \frac{|\mathsf{ext}(C)|}{|C|} = \sum_{C \in \mathcal{C}} |\mathsf{ext}(C)| = \sum_{x \in V(G/\mathcal{C})} \mathsf{edeg}_{G/\mathcal{C}}(x) = 2 \cdot |E(G/\mathcal{C})|$.

For the lower bound, let X be a c-bond cover of G and let $F = V(G) \setminus X$, $\mathcal{C}_X = \{C \in \mathcal{C} \mid C \cap X \neq \emptyset\}$, and $\mathcal{C}_F = \mathcal{C} \setminus \mathcal{C}_X$. Since $\delta(G/\mathcal{C}) \geq 8c$, we obtain that $|E(G/\mathcal{C})|/2 \geq 2c \cdot |V(G/\mathcal{C})| = 2c \cdot |\mathcal{C}|$. We claim that $\sum_{C \in \mathcal{C}_X} |\mathsf{ext}(C)| \geq |E(G/\mathcal{C})|/2$. Indeed, if this is not the case then, by the fact that $|E(G/\mathcal{C})| \leq |E(G[F]/\mathcal{C}_F)| + \sum_{C \in \mathcal{C}_X} |\mathsf{ext}(C)|$, we have that $|E(G[F]/\mathcal{C}_F)| > |E(G/\mathcal{C})|/2 \geq 2c \cdot |\mathcal{C}| \geq 2c \cdot |\mathcal{C}_F|$ and this last inequality, gives that θ_c is a minor of $G/\mathcal{C}_F$ which is a minor of $G[F]$, a contradiction. Here we use the fact that for every θ_c-minor free multigraph G, it holds that $|E(G)| \leq 2c \cdot |V(G)|$ (following from the main combinatorial result of [36]). Therefore, since each set in $\mathcal{C}_X$ contains at least one vertex of X, we obtain $\sum_{v \in X} w(v) \geq \sum_{C \in \mathcal{C}_X} \frac{|\mathsf{ext}(C)|}{|C|} \geq \frac{1}{r} \sum_{C \in \mathcal{C}_X} |\mathsf{ext}(C)| \geq \frac{|E(G/\mathcal{C})|}{2r}$, which proves the lower bound. ◀

Replacing outgrowths. A *c-outgrowth replacer* (hereinafter *replacer*) is a uniformly polynomial-time algorithm which, given a weighted graph $G = (V, E, w)$ and a c-outgrowth $\mathbf{K} = (K, u, v)$ of size at least c, outputs a weighted graph $G' = (V', E', w')$ with the following property.

1. $\mathbf{K}$ is replaced by another c-outgrowth $\mathbf{K}' = (K', u, v)$ of size at most $c - 1$.
2. $\mathsf{opt}(G) = \mathsf{opt}(G')$.
3. Given a c-bond cover $S' \subseteq V(G')$, one can construct in polynomial time a c-bond cover $S \subseteq V(G)$ such that $w(S) \leq w(S')$.

We now present our c-outgrowth replacer. Given a w-weighted graph G, we denote by $\mathsf{opt}(G)$ the weight of an optimal solution for Weighted c-Bond Cover on G.

▶ **Lemma 4.** *For every positive $c \in \mathbb{N}$, there is a c-outgrowth replacer. In particular, an α-approximate solution for G' implies an α-approximate solution for G.*

Proof. For $i \in \{0, \ldots, c-1\}$, let $K_i^{(u,v)}$ be the graph obtained from $K^{(u,v)}$ by adding i edges connecting u and v. Obviously $K_0^{(u,v)}$ equals $K^{(u,v)}$. Let also $T_i \subseteq V(K)$ be a minimum weight set contained in $V(K)$ such that $K_i^{(u,v)} - T_i$ is θ_c-minor-free, and $w_i = w(T_i)$. Note that $T_i \subseteq V(K)$ implies that T_i contains neither u nor v. For example, T_{c-1} is a minimum (internal) vertex cut separating u and v in $K^{(u,v)}$, and $w_{c-1} = w(T_{c-1})$ is finite since there is no edge between u and v in $K^{(u,v)}$. By definition, it holds that $0 = w_0 \leq w_1 \leq \cdots \leq w_{c-1} < \infty$, and these values can be computed in uniformly polynomial time by using dynamic programming on θ_c-minor free graphs (that is bounded treewidth graphs). We also remark that T_j is a c-bond cover of $K_i^{(u,v)}$ for all $i \leq j$. We construct the c-outgrowth $\mathbf{K}' = (K', u, v)$ so that $K'^{(u,v)}$ is as follows (see Figure 1).

- $V(K'^{(u,v)}) = \{u, v, x_1, \ldots, x_{c-1}\}$ where $K' = \{x_1, \ldots, x_{c-1}\}$. For each $1 \leq i \leq c-1$, the weight of x_i is w_i.
- There are edges $(u, x_1), (x_1, x_2), \ldots, (x_{c-2}, x_{c-1}), (x_{c-1}, v)$. Additionally for each $2 \leq i \leq c-1$, there is an edge (x_i, u).

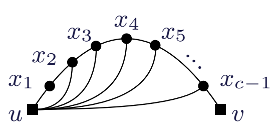

Figure 1 The construction of the replacement c-outgrowth $\mathbf{K}' = (K', u, v)$.

We observe that for each $i \in \{0, \ldots, c-1\}$, the set $\{x_i\}$ is the minimum weight c-bond cover of $K_i'^{(u,v)}$. The next claims are handy (the proof is omitted in this extended abstract).

▷ Claim 5. Let (K,u,v) be a c-outgrowth in G and let $M=(X,Y)$ be a minimal θ_c-model in G. If M does not contain u, then we have $(X\cup Y)\cap V(K)=\emptyset$. Furthermore, if S is a minimal c-bond cover of G and if S contains u or v, say u, then $S\cap V(K)=\emptyset$.

▷ Claim 6. Let Z be a c-bond cover of $G-V(K)$ and let ℓ be the maximum integer[6] such that $G-(K\cup Z)$ contains a θ_ℓ-model with u and v in different sets. Then $Z'=Z\cup T_\ell$ is a c-bond cover of G.

We begin with proving the third condition of the replacer. Let G' be the graph where $K^{(u,v)}$ is replaced by $K'^{(u,v)}$. It suffices to prove the second statement for an arbitrary *minimal* c-bond cover $S'\subseteq V(G')$ of G'.

First, assume that S' contains u or v, say u. Claim 5 is applied to G' verbatim with $G\leftarrow G'$, $K\leftarrow K'$, $K^{(u,v)}\leftarrow K'^{(u,v)}$, and we deduce that $S'\cap V(K')=\emptyset$. Now we take $S\leftarrow S'$, and let us argue that S is a c-bond cover of G. Again Claim 5 implies that if $G-S$ contains a θ_c-model, then one can find one disjoint from $V(K)$. This is not possible because $S=S'$ is a c-bond cover of $G-K=G'-K'$.

Secondly, let us assume that $S'\cap\{u,v\}=\emptyset$. Let ℓ be the maximum integer such that $G'-(K'\cup S')$ contains a θ_ℓ-model $M=(X,Y)$ with u and v in different sets, say $u\in X$ and $v\in Y$. Clearly ℓ is strictly smaller than c because S' is a c-bond cover of $G'-K'$. Note that $K_\ell'^{(u,v)}$ is obtained from $G'[X\cup Y\cup V(K')]$ by contracting X and Y. Because $S'\cap V(K')$ is a c-bond cover of $G'[X\cup Y\cup V(K')]$, it is also a c-bond cover of $K_\ell'^{(u,v)}$. Therefore we have $w_\ell\le w(S'\cap V(K'))$.

Let $S=(S'\setminus V(K'))\cup T_\ell$ be a vertex set of G and note that $w(S)\le w(S')$. Now applying Claim 6 to G with $Z\leftarrow S'\setminus V(K')$ (as a vertex set of G), we conclude that S is a c-bond cover of G. This proves the third condition of the replacer, which also establishes $\mathsf{opt}(G)\le\mathsf{opt}(G')$ in the second condition of the replacer.

It remains to show $\mathsf{opt}(G)\ge\mathsf{opt}(G')$. Consider an optimal c-bond cover S of G, and let p be the maximum integer such that $G-(K\cup S)$ contains a θ_p-model $M=(X,Y)$ with u and v in different sets. Again we apply Claim 6 with $G\leftarrow G'$, $Z\leftarrow S\setminus V(K)$ (as a vertex set of G'), $K\leftarrow K'$ and $T_\ell\leftarrow\{x_p\}$, and derive that $(S\setminus V(K))\cup\{x_p\}$ is a c-bond cover of G'. Lastly, observe that $K_p^{(u,v)}$ is a minor of $G[X\cup Y\cup V(K)]$, and because $S\cap V(K)$ is a c-bond cover of the latter, it is also a c-bond cover of the former. Therefore, we have $w(S\cap V(K))\ge w_p$, from which we have $\mathsf{opt}(G)=w(S)\ge w(S\setminus V(K))+w_p=w((S\setminus V(K))\cup\{x_p\})\ge\mathsf{opt}(G')$. ◀

3 The primal-dual approach

We begin the section by formalizing the notion of α-thin layer. An *α-thin layer* of a weighted graph $G=(V,E,w)$ is a weighted graph $H=(V,E,w^o)$ such that the following holds.

- $w^o(v)\le w(v)$ for every $v\in V$,
- $w^o(v)=w(v)$ for some $v\in V$, and
- $w^o(S)\ge(1/\alpha)\cdot w^o(V)$ for any c-bond cover $S\subseteq V$ of H.

We are now ready to prove our main approximation result.

▶ **Theorem 7.** *There is a uniformly polynomial-time algorithm which, given a positive integer c and a weighted graph $G=(V,E,w)$, computes a c-bond cover of weight at most $\alpha\cdot\mathsf{opt}(G)$ for some $\alpha=\alpha(c)$.*

[6] If u and v are not connected in $G-(K\cup Z)$, we let $\ell=0$.

Proof. The algorithm initially sets $G_1 = G$, and iteratively constructs a sequence of weighted graphs $G_i = (V_i, E_i, w_i)$ for $i = 0, 1, \ldots$. At i-th iteration, we run the algorithm $\mathcal{A}$ of Theorem 2 for $t = 8c$. If $\mathcal{A}$ detects a c-outgrowth of size at least c, then we call the algorithm of Lemma 4, which is clearly a replacer. We run the replacer on G_i and set G_{i+1} to be the output of the replacer. If a θ_c-model M of G_i of size at most $f_1(c,t)$ is detected by $\mathcal{A}$, then let $\epsilon := \min\{w_i(v) : v \in M\}$ and consider the weighted graph $H_i = (V_i, E_i, w_i^o)$ with the weight function w as follows:

$$w_i^o(v) = \begin{cases} \epsilon & \text{if } v \in M \\ 0 & \text{otherwise.} \end{cases}$$

It is obvious that H_i is an α-thin layer with $\alpha = f_1(c,t)$.

In the third case, note that the cluster collection $\mathcal{C}$ forms a cluster partition of $G_i[\bigcup \mathcal{C}]$. Consider the weight function $w : \bigcup \mathcal{C} \to \mathbb{R}_{\geq 0}$ as in Equation 1 of $G_i[\bigcup \mathcal{C}]$. Let $\epsilon := \min\{w_i(v)/w(v) : v \in \bigcup \mathcal{C}\}$ and $H_i = (V_i, E_i, w_i^o)$ be the weighted graph, where

$$w_i^o(v) = \begin{cases} \epsilon \cdot w(v) & \text{if } v \in \bigcup \mathcal{C} \\ 0 & \text{otherwise.} \end{cases}$$

Let us verify that H_i is an α-thin layer of G_i for $\alpha = 4r$, where $r = f_1(c,t)$. It is straightforward to see that the first two requirement of α-thin layer are met due to the choice of ϵ. To check the last requirement, consider an arbitrary c-bond cover $S \subseteq V_i$ of H_i. By Lemma 3, it holds that

$$\frac{\epsilon}{2r} \cdot |E(G_i[\bigcup \mathcal{C}]/\mathcal{C})| \leq \sum_{v \in S} w_i^o(v) \leq \sum_{v \in V_i} w_i^o(v) \leq 2\epsilon \cdot |E(G_i[\bigcup \mathcal{C}]/\mathcal{C})|,$$

and therefore,

$$\sum_{v \in S} w_i^o(v) \geq \frac{\epsilon}{2r} \cdot |E(G_i[\bigcup \mathcal{C}]/\mathcal{C})| \geq \frac{1}{4r} \cdot \sum_{v \in V_i} w_i^o(v).$$

In both the second and the third cases, we set G_{i+1} to be the weighted graph $(V_i, E_i, w_i - w_i^o)$ after removing all vertices of weight zero.

Finally, if $\mathcal{A}$ reports that G_i is θ_c-minor free, then we terminate the iteration. Let $G = G_1, G_2, \ldots, G_\ell$ be the constructed sequence of weighted graph at the end, with G_ℓ being a θ_c-minor-free graph. Observe that our algorithm strictly decrease the number of vertices before the ℓ-th iteration, and thus $\ell \leq n$.

To establish the main statement, it suffices show that there is a polynomial-time algorithm which produces an $4r$-approximate solution for G_i given an $4r$-approximate solution T_{i+1} for G_{i+1}, where $r = f_1(c,t)$ and $t = 8c$. This trivially holds if the execution of $\mathcal{A}$ at i-th iteration calls the replacer.

Suppose that i-th iteration produces an α-thin layer $H_i = (V_i, E_i, w_i^o)$, and recall that every α-thin layer produced in our algorithm satisfies $\alpha \leq 4r$. As T_{i+1} is an $4r$-approximate solution for G_{i+1}, we have

$$\mathsf{opt}(G_{i+1}) \quad \geq \quad (1/4r) \cdot w_{i+1}(T_{i+1}), \tag{2}$$

▷ Claim 8. $T_i := T_{i+1} \cup (V_i \setminus V_{i+1})$ is an $4r$-approximate solution for G_i.

Proof. Let $D_i = V_i \setminus V_{i+1}$, namely the vertices deleted from G_i to obtain G_{i+1}. It is obvious that $T_{i+1} \cup D_i$ is a feasible solution for G_i, that is, a c-bond cover of G_i because $G_{i+1} - T_{i+1} = G_i - (T_{i+1} \cup D_i)$ and T_{i+1} is a c-bond cover of G_{i+1}. Let $Q \subseteq V_i$ be an optimal solution for G_i. Then Q is a feasible solution for H_i and $Q \cap V_{i+1}$ is a feasible solution for G_{i+1}, therefore

$$w_i^o(Q) \geq (1/4r) \cdot w_i^o(V_i) \text{ and} \tag{3}$$
$$w_{i+1}(Q \cap V_{i+1}) \geq \mathsf{opt}(G_{i+1}), \tag{4}$$

where the inequality 3 is due to the third requirement of α-thin layer. Furthermore, it holds that

$$w_i(v) = w_i^o(v) + w_{i+1}(v) \quad \text{for each } v \in V_{i+1} \text{ and} \tag{5}$$
$$w_i(v) = w_i^o(v) \quad \text{for each } v \in D_i. \tag{6}$$

Therefore,

$$\begin{aligned} w_i(Q) &= w_i^o(Q) + w_{i+1}(Q \cap V_{i+1}) && \because (5), (6) \\ &\geq (1/4r) \cdot w_i^o(V_i) + \mathsf{opt}(G_{i+1}) && \because (3), (4) \\ &\geq (1/4r) \cdot w_i^o(T_{i+1} \cup D_i) + (1/4r) \cdot w_{i+1}(T_{i+1}) && \because (2) \\ &= (1/4r) \cdot (w_i^o(T_{i+1}) + w_{i+1}(T_{i+1})) + (1/4r) \cdot w_i^o(D_i) \\ &= (1/4r) \cdot w_i(T_{i+1} \cup D_i) && \because (5), (6) \end{aligned}$$

and the claim follows. ◁

We inductively obtain a $4r$-approximate solution for G_i, and finally for the graph $G_1 = G$. This finishes the proof. ◀

4 Discussion

In this paper we construct a polynomial-time constant-factor approximation algorithm for the Weighted $\mathcal{F}$-Vertex Deletion problem in the case $\mathcal{F}$ is the class of graphs not containing a c-bond or, alternatively, the θ_c-minor free graphs. The constant-factor of our approximation algorithm is a (constructible) function of c and the running time is uniformly polynomial. Our results, in case $c = 2$, yield a constant-factor approximation for the Weighted Feedback Vertex Set. Also, a constant-factor approximation for Weighted Diamond Hitting Set can easily be derived for the case where $c = 3$. For this we apply our results on simple graphs and observe that each time a θ_3-minor-model appears, this model, under the absence of multiple edges, should contain 4 vertices and therefore is a minor-model of the diamond K_4^- (that is K_4 without an edge).

Certainly the general open question is whether Weighted $\mathcal{F}$-Vertex Deletion admits a constant-factor approximation for more general instantiations of the minor-closed class $\mathcal{F}$. In this direction, the challenge is to use our approach when the graphs in $\mathcal{F}$ have bounded treewidth (or, equivalently, if the minor obstruction of $\mathcal{F}$ contains some planar graph). For this, one needs to extend the structural result of Theorem 2 and, based on this to build a replacer as in Lemma 4.

Given an $r \in \mathbb{N}$, an *r-protrusion* of a graph G is a set $X \subseteq V(G)$ such that $G[X]$ has treewidth at most t and $|\partial_G(X)| \leq t$, were $\partial_G(X)$ is the set of vertices of X that are incident to edges not in $G[X]$. We conjecture that a possible extension of Theorem 2 might be the following.

▶ **Conjecture 9.** *There are functions $f_2 : \mathbb{N}^2 \to \mathbb{N}$ and $f_3 : \mathbb{N}^3 \to \mathbb{N}$ such that, for every h-vertex planar graph H and every two positive integers t, p, there is a uniformly polynomial time algorithm that, given as input a graph G, outputs one of the following:*

1. *an $f_2(h,t)$-protrusion X of size at least p, or*
2. *a minor-model of H of of size at most $f_3(h,t,p)$, or*
3. *a cluster collection $\mathcal{C}$ of G of capacity at most $f_3(h,t,p)$ such that $\delta(G/\mathcal{C}) \geq t$, or*
4. *a report that G is H-minor free.*

Given a proof of some suitable version of Conjecture 9 at hand, cases 1,2, and 3 above can be treated using the method proposed in this paper. In the first case, we need to find a *weighted prorusion replacer* that can replace, in the weighed graph $G = (V, E, w)$, the subgraph $G[X]$ by another one (glued on the same boundary) and create a new weighted graph $G' = (V', E', w')$ so that an optimal solution has the same weight in both instances. In our case, the role of a protrusion is played by the c-outgrowth, where X is the vertex set of $K^{(u,v)}$ that has treewidth at most $2c$ and $|\partial_G(X)| \leq 2$, i.e., $V(K^{(u,v)})$ is a $2c$-protrusion of G. In the case of θ_c, the the replacer is given in Lemma 4. The existence of such a replacer in the general case is wide open, first because the boundary $\partial_G(X)$ has bigger size (depending on h but perhaps also on t) and second, and most important, because we now must deal with *weights* which does not permit us to use any protrusion replacement machinery such as the one used in [20, 19] unweighted version of the problem (based on the, so called, FII-property [13] for more details).

We believe that a possible way to prove Conjecture 9 is to use as departure the proof of the main combinatorial result in [45]. However, in our opinion, the most challenging step is to design a weighted protrusion replacer (or, on the negative side, to provide instantiations of H where such a replacer does not exist). As such a replacer needs to work on the presence of weights, we suggest that its design might use techniques related to mimicking networks technology [24, 34].

Finally, since our algorithm is based on the primal-dual framework and proceeds by constructing suitable weights for the second and third case where every feasible solution is $\mathcal{O}(1)$-approximate, one can ask whether it is possible to *bypass* the need for a replacer and construct suitable weights for the first case. Indeed, the previous approximation algorithms for Weighted Feedback Vertex Set [7, 12, 16] designed suitable weights even for the case 1 where every *minimal* solution is $\mathcal{O}(1)$-approximate. (And used the additional "reverse delete" step at the end to ensure that the final solution remains minimal, for every weighted graph constructed.) In the full version of the paper [32], we show that such weights *cannot exist* for a simple planar graph H, which suggests that replacers are inherently needed for this class of algorithms for Weighted $\mathcal{F}$-Vertex Deletion.

References

1 Akanksha Agrawal, Sudeshna Kolay, Daniel Lokshtanov, and Saket Saurabh. A faster FPT algorithm and a smaller kernel for block graph vertex deletion. In *LATIN 2016: theoretical informatics*, volume 9644 of *Lecture Notes in Comput. Sci.*, pages 1–13. Springer, Berlin, 2016. doi:10.1007/978-3-662-49529-2_1.

2 Akanksha Agrawal, Daniel Lokshtanov, Pranabendu Misra, Saket Saurabh, and Meirav Zehavi. Polylogarithmic approximation algorithms for weighted-$\mathcal{F}$-deletion problems. In *Approximation, randomization, and combinatorial optimization. Algorithms and techniques*, volume 116 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 1, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2018.

3 Akanksha Agrawal, Daniel Lokshtanov, Pranabendu Misra, Saket Saurabh, and Meirav Zehavi. Feedback vertex set inspired kernel for chordal vertex deletion. *ACM Trans. Algorithms*, 15(1):Art. 11, 28, 2019. doi:10.1145/3284356.

4 Akanksha Agrawal, Pranabendu Misra, Saket Saurabh, and Meirav Zehavi. Interval vertex deletion admits a polynomial kernel. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1711–1730. SIAM, Philadelphia, PA, 2019. doi:10.1137/1.9781611975482.103.

5 Jungho Ahn, Eduard Eiben, O-joung Kwon, and Sang-il Oum. A polynomial kernel for 3-leaf power deletion. In *45th International Symposium on Mathematical Foundations of Computer Science*, volume 170 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages 5:1–5:14. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2020.

6 Jungho Ahn, Eun Jung Kim, and Euiwoong Lee. Towards constant-factor approximation for chordal / distance-hereditary vertex deletion. In Yixin Cao, Siu-Wing Cheng, and Minming Li, editors, *31st International Symposium on Algorithms and Computation, ISAAC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, volume 181 of *LIPIcs*, pages 62:1–62:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ISAAC.2020.62.

7 Vineet Bafna, Piotr Berman, and Toshihiro Fujito. A 2-approximation algorithm for the undirected feedback vertex set problem. *SIAM J. Discret. Math.*, 12(3):289–297, 1999. doi:10.1137/S0895480196305124.

8 Nikhil Bansal, Daniel Reichman, and Seeun William Umboh. LP-based robust algorithms for noisy minor-free and bounded treewidth graphs. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1964–1979. SIAM, Philadelphia, PA, 2017. doi:10.1137/1.9781611974782.128.

9 Reuven Bar-Yehuda, Keren Bendel, Ari Freund, and Dror Rawitz. Local ratio: A unified framework for approximation algorithms. in memoriam: Shimon even 1935-2004. *ACM Computing Surveys (CSUR)*, 36(4):422–463, 2004.

10 Reuven Bar-Yehuda and Simon Even. A local-ratio theorem for approximating the weighted vertex cover problem. In G. Ausiello and M. Lucertini, editors, *Analysis and Design of Algorithms for Combinatorial Problems*, volume 109 of *North-Holland Mathematics Studies*, pages 27–45. North-Holland, 1985. doi:10.1016/S0304-0208(08)73101-3.

11 Julien Baste, Ignasi Sau, and Dimitrios M. Thilikos. A complexity dichotomy for hitting connected minors on bounded treewidth graphs: the chair and the banner draw the boundary. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 951–970. SIAM, 2020. doi:10.1137/1.9781611975994.57.

12 Ann Becker and Dan Geiger. Optimization of pearl's method of conditioning and greedy-like approximation algorithms for the vertex feedback set problem. *Artificial Intelligence*, 83(1):167–188, 1996. doi:10.1016/0004-3702(95)00004-6.

13 Hans L. Bodlaender, Fedor V. Fomin, Daniel Lokshtanov, Eelko Penninkx, Saket Saurabh, and Dimitrios M. Thilikos. (meta) kernelization. *J. ACM*, 63(5):44:1–44:69, 2016. doi:10.1145/2973749.

14 Yixin Cao. Linear recognition of almost interval graphs. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1096–1115. ACM, New York, 2016. doi:10.1137/1.9781611974331.ch77.

15 Robert D. Carr, Lisa Fleischer, Vitus J. Leung, and Cynthia A. Phillips. Strengthening integrality gaps for capacitated network design and covering problems. In David B. Shmoys, editor, *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, January 9-11, 2000, San Francisco, CA, USA*, pages 106–115. ACM/SIAM, 2000. URL: http://dl.acm.org/citation.cfm?id=338219.338241.

16 Fabián A. Chudak, Michel X. Goemans, Dorit S. Hochbaum, and David P. Williamson. A primal-dual interpretation of two 2-approximation algorithms for the feedback vertex set problem in undirected graphs. *Oper. Res. Lett.*, 22(4-5):111–118, 1998. doi:10.1016/S0167-6377(98)00021-2.

17 Hiroshi Eto, Tesshu Hanaka, Yasuaki Kobayashi, and Yusuke Kobayashi. Parameterized algorithms for maximum cut with connectivity constraints. In Bart M. P. Jansen and Jan Arne Telle, editors, *14th International Symposium on Parameterized and Exact Computation, IPEC 2019, September 11-13, 2019, Munich, Germany*, volume 148 of *LIPIcs*, pages 13:1–13:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.IPEC.2019.13.

18 Samuel Fiorini, Gwenaël Joret, and Ugo Pietropaoli. Hitting diamonds and growing cacti. In *Integer Programming and Combinatorial Optimization – IPCO 2010*, volume 6080 of *Lecture Notes in Comput. Sci.*, pages 191–204. Springer, Berlin, 2010. doi:10.1007/978-3-642-13036-6_15.

19 Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, Geevarghese Philip, and Saket Saurabh. Hitting forbidden minors: Approximation and kernelization. *SIAM J. Discret. Math.*, 30(1):383–410, 2016. doi:10.1137/140997889.

20 Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. Planar $\mathcal{F}$-Deletion: approximation, kernelization and optimal FPT algorithms. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science – FOCS 2012*, pages 470–479. IEEE Computer Soc., Los Alamitos, CA, 2012.

21 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Hitting topological minors is FPT. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 1317–1326. ACM, 2020. doi:10.1145/3357713.3384318.

22 Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. Approximation schemes via width/weight trade-offs on minor-free graphs. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 2299–2318. SIAM, 2020. doi:10.1137/1.9781611975994.141.

23 Anupam Gupta, Euiwoong Lee, Jason Li, Pasin Manurangsi, and MichałWł odarczyk. Losing treewidth by separating subsets. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1731–1749. SIAM, Philadelphia, PA, 2019. doi:10.1137/1.9781611975482.104.

24 Torben Hagerup, Jyrki Katajainen, Naomi Nishimura, and Prabhakar Ragde. Characterizing multiterminal flow networks and computing flows in networks of small treewidth. *J. Comput. Syst. Sci.*, 57(3):366–375, 1998. doi:10.1006/jcss.1998.1592.

25 David J. Haglin and Shankar M. Venkatesan. Approximation and intractability results for the maximum cut problem and its variants. *IEEE Trans. Computers*, 40(1):110–113, 1991. doi:10.1109/12.67327.

26 Ken ichi Kawarabayashi, Yusuke Kobayashi, and Bruce Reed. The disjoint paths problem in quadratic time. *Journal of Combinatorial Theory, Series B*, 102(2):424–435, 2012. doi:10.1016/j.jctb.2011.07.004.

27 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.

28 Bart M. P. Jansen and Marcin Pilipczuk. Approximation and kernelization for chordal vertex deletion. *SIAM J. Discrete Math.*, 32(3):2258–2301, 2018. doi:10.1137/17M112035X.

29 Gwenaël Joret, Christophe Paul, Ignasi Sau, Saket Saurabh, and Stéphan Thomassé. Hitting and harvesting pumpkins. *SIAM J. Discret. Math.*, 28(3):1363–1390, 2014. doi:10.1137/120883736.

30 Ken-ichi Kawarabayashi and Anastasios Sidiropoulos. Polylogarithmic approximation for minimum planarization (almost). In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 779–788. IEEE Computer Society, 2017. doi:10.1109/FOCS.2017.77.

31 Eun Jung Kim, Alexander Langer, Christophe Paul, Felix Reidl, Peter Rossmanith, Ignasi Sau, and Somnath Sikdar. Linear kernels and single-exponential algorithms via protrusion decompositions. *ACM Trans. Algorithms*, 12(2):21:1–21:41, 2016. doi:10.1145/2797140.

32 Eun Jung Kim, Euiwoong Lee, and Dimitrios M. Thilikos. A constant-factor approximation for weighted bond cover, 2021. arXiv:2105.00857.

33 Stefan Kratsch and Magnus Wahlström. Compression via matroids: a randomized polynomial kernel for odd cycle transversal. *ACM Trans. Algorithms*, 10(4):Art. 20, 15, 2014. doi:10.1145/2635810.

34 Robert Krauthgamer and Inbal Rika. Mimicking networks and succinct representations of terminal cuts. In Sanjeev Khanna, editor, *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 1789–1799. SIAM, 2013. doi:10.1137/1.9781611973105.128.

35 John M. Lewis and Mihalis Yannakakis. The node-deletion problem for hereditary properties is NP-complete. *J. Comput. System Sci.*, 20(2):219–230, 1980. ACM-SIGACT Symposium on the Theory of Computing (San Diego, Calif., 1978).

36 Stratis Limnios, Christophe Paul, Joanny Perret, and Dimitrios M. Thilikos. Edge degeneracy: Algorithmic and structural results. *Theor. Comput. Sci.*, 839:164–175, 2020. doi:10.1016/j.tcs.2020.06.006.

37 Daniel Lokshtanov, Pranabendu Misra, Fahad Panolan, Geevarghese Philip, and Saket Saurabh. A $(2 + \epsilon)$-factor approximation algorithm for split vertex deletion. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 80:1–80:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ICALP.2020.80.

38 Anand Louis, Prasad Raghavendra, and Santosh Vempala. The complexity of approximating vertex expansion. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 360–369. IEEE, 2013.

39 Carsten Lund and Mihalis Yannakakis. The approximation of maximum subgraph problems. In Andrzej Lingas, Rolf G. Karlsson, and Svante Carlsson, editors, *Automata, Languages and Programming, 20nd International Colloquium, ICALP93, Lund, Sweden, July 5-9, 1993, Proceedings*, volume 700 of *Lecture Notes in Computer Science*, pages 40–51. Springer, 1993. doi:10.1007/3-540-56939-1_60.

40 George L. Nemhauser and Leslie E. Trotter Jr. Properties of vertex packing and independence system polyhedra. *Math. Program.*, 6(1):48–61, 1974. doi:10.1007/BF01580222.

41 Bruce Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Oper. Res. Lett.*, 32(4):299–301, 2004. doi:10.1016/j.orl.2003.10.009.

42 Neil Robertson and Paul D. Seymour. Graph minors. V. excluding a planar graph. *J. Comb. Theory, Ser. B*, 41(1):92–114, 1986. doi:10.1016/0095-8956(86)90030-4.

43 Neil Robertson and Paul D. Seymour. Graph minors. XX. Wagner's conjecture. *J. Comb. Theory, Ser. B*, 92(2):325–357, 2004. doi:10.1016/j.jctb.2004.08.001.

44 Ignasi Sau, Giannos Stamoulis, and Dimitrios M. Thilikos. An fpt-algorithm for recognizing k-apices of minor-closed graph classes. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 95:1–95:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ICALP.2020.95.

45 Wouter Cames van Batenburg, Tony Huynh, Gwenaël Joret, and Jean-Florent Raymond. A tight erdős-pósa function for planar minors. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1485–1500. SIAM, 2019. doi:10.1137/1.9781611975482.90.