A New Framework for Kernelization Lower Bounds: The Case of Maximum Minimal Vertex Cover

Júlio Araújo
Departament of Mathematics, Federal University of Ceará, Fortaleza, Brazil

Marin Bougeret
LIRMM, Université de Montpellier, France

Victor Campos
Departament of Computer Science, Federal University of Ceará, Fortaleza, Brazil

Ignasi Sau
LIRMM, Université de Montpellier, CNRS, France

Abstract

In the Maximum Minimal Vertex Cover (MMVC) problem, we are given a graph G and a positive integer k, and the objective is to decide whether G contains a minimal vertex cover of size at least k. Motivated by the kernelization of MMVC with parameter k, our main contribution is to introduce a simple general framework to obtain lower bounds on the degrees of a certain type of polynomial kernels for vertex-optimization problems, which we call lop-*kernels*. Informally, this type of kernels is required to preserve large optimal solutions in the reduced instance, and captures the vast majority of existing kernels in the literature. As a consequence of this framework, we show that the trivial quadratic kernel for MMVC is essentially optimal, answering a question of Boria et al. [Discret. Appl. Math. 2015], and that the known cubic kernel for Maximum Minimal Feedback Vertex Set is also essentially optimal. On the positive side, given the (plausible) non-existence of subquadratic kernels for MMVC on general graphs, we provide subquadratic kernels on H-free graphs for several graphs H, such as the bull, the paw, or the complete graphs, by making use of the Erdős-Hajnal property in order to find an appropriate decomposition. Finally, we prove that MMVC does not admit polynomial kernels parameterized by the size of a minimum vertex cover of the input graph, even on bipartite graphs, unless $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$. This indicates that parameters smaller than the solution size are unlike to yield polynomial kernels for MMVC.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Maximum minimal vertex cover, parameterized complexity, polynomial kernel, kernelization lower bound, Erdős-Hajnal property, induced subgraphs

Digital Object Identifier 10.4230/LIPIcs.IPEC.2021.4

Related Version *Full Version*: https://arxiv.org/abs/2102.02484

Funding *Júlio Araújo*: CNPq-Pq 304478/2018-0, CAPES-PrInt 88887.466468/2019-00 and CAPES-STIC-AmSud 88881.569474/2020-01.
Victor Campos: FUNCAP - PNE-011200061.01.00/16.
Ignasi Sau: DEMOGRAPH (ANR-16-CE40-0028), ESIGMA (ANR-17-CE23-0010), ELIT (ANR-20-CE48-0008-01), and French-German Collaboration ANR/DFG Project UTMA (ANR-20-CE92-0027).

Acknowledgements We would like to thank Michael Lampis (resp. Magnus Wahlström, Venkatesh Raman) for pointing us to reference [27] (resp. reference [34], references [7, 44]).

1 Introduction

A *vertex cover* in a graph G is a subset of vertices containing at least one endpoint of every edge. In the associated optimization problem, called Minimum Vertex Cover, the objective is to find, given an input graph G, a vertex cover in G of minimum size. This

16th International Symposium on Parameterized and Exact Computation (IPEC 2021).
Editors: Petr A. Golovach and Meirav Zehavi; Article No. 4; pp. 4:1–4:19
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

problem has been one of the leitmotifs of the area of parameterized complexity [21,26], serving as a test bed for many of the most fundamental techniques. An instance of a *parameterized problem* is of the form (x, k), where x is the total input (typically, a graph) and k is a positive integer called the *parameter*. The crucial notion is that of *fixed-parameter tractable* algorithm, FPT for short, which is an algorithm deciding whether (x, k) is a positive instance in time $f(k) \cdot |x|^{\mathcal{O}(1)}$, where f is a computable function depending only on k. In the parameterized Vertex Cover problem, we are given a graph G and an integer parameter k, and the objective is to decide whether G contains a vertex cover of size at most k. One of the main fields within parameterized complexity is *kernelization* [31], where the objective is to decide whether an instance (x, k) of a parameterized problem can be transformed in polynomial time into an equivalent instance (x', k') whose total size is bounded by a function of k; the reduced instance is called a *kernel*, and finding kernels of small size, typically polynomial or even linear in k in the best case, is one of the most active areas of parameterized complexity. There are several techniques for obtaining linear kernels for the Vertex Cover problem [31].

Considering the "max-min" version of minimization problems, that is, maximizing the size of a *minimal* solution of the corresponding problem, is a natural approach that has been applied to several problems such as Dominating Set [5,28] (whose "max-min" version is called Upper Domination), Feedback Vertex Set [27], or Hitting Set [4,22]. In this article we are interested in the "max-min" version of Minimum Vertex Cover, called Maximum Minimal Vertex Cover, or just MMVC for short.

Previous work. In his habilitation, Fernau [30] presented FPT algorithms for MMVC as well as some results about its kernelization parameterized by the solution size k. It is easy to note, as observed in [30], that the problem admits a kernel with at most k^2 vertices: if some vertex has degree at least k, we can safely answer "yes" (cf. Lemma 2 for a proof); otherwise, the maximum degree is at most $k - 1$, and it follows that every instance without isolated vertices (which may be safely removed) that has at least k^2 vertices is a yes-instance, hence we have a trivial kernel with at most k^2 vertices. Fernau [30] presented a kernel with at most $4k$ vertices for MMVC restricted to planar instances using the algorithmic version of the Four Color Theorem [47], and claimed in [30, Corollary 4.25] a kernel with at most $2k$ vertices on general graphs using spanning trees. Unfortunately, this latter kernelization algorithm is incorrect, as we discuss at the end of Section 3.

Boria et al. [16] initiated a study of the complexity of MMVC and presented a number of results, in particular a polynomial-time approximation algorithm with ratio $n^{1/2}$ on n-vertex graphs, and showed that, unless P = NP, no polynomial-time approximation algorithm with ratio $n^{1/2-\varepsilon}$ exists for any $\varepsilon > 0$. They also presented FPT algorithms for MMVC for several choices of the parameters such as the treewidth, the size of a maximum matching, or the size of a minimum vertex cover of the input graph. The authors asked explicitly whether kernels of size $o(k^2)$ exist for MMVC parameterized by k.

Zehavi [50] presented tight FPT algorithms, under the Strong Exponential Time Hypothesis, for MMVC and its weighted version parameterized by the size of a minimum vertex cover. Recently, Bonnet and Paschos [14] and Bonnet et al. [13] considered the inapproximability of MMVC in subexponential time.

Note that the MMVC problem is the dual of the well-studied Minimum Independent Dominating Set problem (to see this, note that the complement of any minimal vertex cover is an independent dominating set), which has applications in wireless and ad-hoc networks [42]. We refer to the survey of Goddard and Henning [35].

Our results and techniques. In this article we focus on the kernelization of the MMVC problem, which has been almost unexplored so far in the literature. Motivated by the question of Boria et al. [16] about the existence of subquadratic kernels for MMVC, we introduce a generic framework to obtain lower bounds on the degree of a "certain type" of polynomial kernels for parameterized vertex-maximization problems (in particular, for MMVC), based on a hypothesis that guarantees an inapproximability result, typically $\mathsf{P} \neq \mathsf{NP}$. Informally, by "certain type" we mean kernelization algorithms that, in polynomial time, either decide the instance (by answering "yes" or "no") or produce an equivalent instance of the considered problem in which the value of an optimal solution is "preserved", in the sense that it may drop only by the drop suffered by the parameter; see Definition 6 for the formal details. We call such kernels *large optimal preserving kernels*, or lop-*kernels* for short. Even if this type of kernels may seem restrictive, we are not aware of any known polynomial kernel for a vertex-maximization problem, such as those that have become nowadays standard [31], which is *not* a lop-kernel. The idea of our approach is to show (Theorem 3) that a lop-kernel yields a polynomial-time approximation algorithm whose ratio depends on the degree of the kernel, and to use known inapproximability results to obtain the desired lower bound.

Combining Theorem 3 (for $r = \frac{1}{2}$) with the known $\mathcal{O}(n^{\frac{1}{2}-\varepsilon})$-inapproximability result for MMVC by Boria et al. [16] immediately rules out the existence of a lop-kernel for MMVC with $\mathcal{O}(k^{2-\varepsilon})$ vertices for any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$. Thus, while Corollary 4 does not completely rule out the existence of subquadratic kernels for MMVC, it tells that, if such a kernel exists, it should consist of "non-standard" reduction rules.

Interestingly, our framework has consequences beyond the MMVC problem, namely for the Maximum Minimal Feedback Vertex Set (MMFVS) problem, defined in the natural way. Dublois et al. [27] recently provided a cubic kernel for MMFVS parameterized by the solution size, and proved that the problem does not admit an $\mathcal{O}(n^{\frac{2}{3}-\varepsilon})$-approximation algorithm for any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$. Hence, by applying Theorem 3 with $r = \frac{2}{3}$ we obtain (Corollary 5) that the cubic kernel of Dublois et al. [27] is essentially optimal.

In Section 4 we translate our framework to vertex-minimization problems, whose applicability is summarized in Theorem 13. Compared to existing frameworks to obtain lower bounds on kernelization, such as cross-compositions [8, 10], weak compositions [23, 24, 40], polynomial parameter transformations [6, 11], or techniques to obtain lower bounds on the coefficients of linear kernels [18], or that relate approximation and kernelization [1,7,37,43,45], our approach has the advantages that it is quite simple, straightforward to apply, and on same hypothesis on which the inapproximability result is based, typically $\mathsf{P} \neq \mathsf{NP}$. On the negative side, it has the following two drawbacks. The first one is that it can only be applied to vertex-maximization (or minimization) problems which are very hard to approximate, namely within a factor $\mathcal{O}(n^{r-\varepsilon})$ for some constant $r > 0$. Finally, our techniques are able to rule out the existence of what we call lop-kernels of certain sizes, but smaller non-standard kernels that do not preserve the value of large optimal solutions might, a priori, still exist. Hence, since our framework seems to be orthogonal to existing ones, we think that it adds to the above list of techniques to obtain kernelization lower bounds.

Coming back to the MMVC problem parameterized by the solution size, given the above negative result on general graphs, we identify graph classes where MMVC is still NP-hard and admits a subquadratic kernel. In particular, we deal with graph classes defined by excluding an *induced* subgraph H that satisfies the *Erdős-Hajnal property* [29], that is, for which there exists a constant $\delta > 0$ such that every H-free graph on n vertices contains either a clique or an independent set of size n^{δ}. In particular, we present a kernel for MMVC with $\mathcal{O}(k^{7/4})$ vertices on the well-studied class of bull-free graphs (Theorem 15), with $\mathcal{O}(k^{\frac{2t-3}{t-1}})$

vertices on K_t-free graphs for every $t \geq 3$, and with $\mathcal{O}(k^{5/3})$ vertices on paw-free graphs. The latter two results can be found in the full version [3]. To the best of our knowledge, this is the first time that the Erdős-Hajnal property is used to obtain polynomial kernels (we would like to note that it was used by Kratsch et al. [44] to obtain kernelization lower bounds).

Our strategy to obtain these subquadratic kernels on H-free graphs is as follows. By the high-degree rule mentioned above, given an instance (G, k), we may assume that the maximum degree of G is at most $k-1$. We find greedily a minimal vertex cover X of G. If $|X| \geq k$ we are done, so we may assume that $|X| \leq k-1$, hence the goal is to bound the size of $S := V(G) \setminus X$. Using that $G[X]$ is also H-free, the Erdős-Hajnal property implies (Lemma 14) that X can be partitioned in polynomial time into a sublinear (in k) number of independent sets and cliques. Since S is an independent set and we may assume that G has no isolated vertices, in order to bound $|S|$ by a subquadratic function of k, it is enough to show that, for each of the sublinearly many cliques or independent sets Y that partition X, its neighborhood in S has size $\mathcal{O}(k)$. This is easy if Y is an independent set: if $|N_S(Y)| \geq k$ we can conclude that (G, k) is a yes-instance (Lemma 2), so we may assume that $|N_S(Y)| \leq k-1$. The case where Y is a clique is more interesting, and we need ad-hoc arguments depending on each particular excluded induced subgraph H. In the full version [3] we also present several positive results for MMVC restricted to other graph classes, such as $K_{1,t}$-free graphs, or graph classes with bounded cliquewidth or chromatic number.

Finally, we show (Theorem 17) that MMVC, parameterized by the size of a minimum vertex cover (or of a maximum matching) of the input graph, does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly, even restricted to bipartite graphs. This result complements the FPT algorithms for MMVC under these parameterizations given by Boria et al. [16] and Zehavi [50], and shows that, in what concerns the existence of polynomial kernels for MMVC, the most natural structural parameters smaller than the solution size are not large enough to yield polynomial kernels (note that the treewidth of any graph is at most one more than its vertex cover number, hence our result rules out the existence of polynomial kernels for MMVC parameterized by treewidth as well). The proof consists of a polynomial parameter transformation from Monotone Sat parameterized by the number of variables. In particular, our reduction yields also the NP-hardness of MMVC on bipartite graphs, which provides an alternative proof to the one of Boliac and Lozin [12] via the NP-hardness of Minimum Independent Dominating Set on bipartite graphs.

Organization. Due to space limitations, some of the contents have been moved to full version [3]. In Section 2 we provide some basic preliminaries about graphs, the MMVC problem, and parameterized complexity. In Section 3 we state our framework to obtain kernelization lower bounds and present its consequences for MMVC and MMFVS. We present in Section 4 our framework for vertex-minimization problems, but due to space constraints we only provide the basic definitions and the main result (Theorem 13). We discuss in Section 5 the flaw in the linear kernel for MMVC claimed by Fernau [30]. Section 6 is devoted to the subquadratic kernel on bull-free graphs, which captures the main ideas of our approach using the Erdős-Hajnal property. Further subquadratic kernels and other positive results for MMVC can be found in the full version [3]. Our reduction to rule out the existence of polynomial kernels for MMVC parameterized by the size of a minimum vertex cover is presented in Section 7. We conclude the article in Section 8 with a discussion and some directions for further research.

2 Preliminaries

Graphs and functions. We use standard graph-theoretic notation, and we refer the reader to [25] for any undefined notation. For an integer $p \geq 1$, we let $[p]$ be the set containing all integers i with $1 \leq i \leq p$. We use $\uplus$ to denote the disjoint union. We will only consider finite undirected graphs without loops nor multiple edges, and we denote an edge between two vertices u and v by $\{u, v\}$. A subgraph H of a graph G is *induced* if H can be obtained from G by deleting a set of vertices $D = V(G) \setminus S$, and we denote $H = G[S]$. A graph G is *H-free* if it does not contain any induced subgraph isomorphic to H. If $\mathcal{H}$ is a collection of graphs, a graph G is *$\mathcal{H}$-free* is it is H-free for every $H \in \mathcal{H}$. For a graph G and a set $S \subseteq V(G)$, we use the notation $G \setminus S = G[V(G) \setminus S]$, and for a vertex $v \in V(G)$, we abbreviate $G \setminus \{v\}$ as $G \setminus v$. A vertex v is *complete* to a set $S \subseteq V(G)$ if v is adjacent to every vertex in S.

The *open* (resp. *closed*) *neighborhood* of a vertex v is denoted by $N(v)$ (resp. $N[v]$), whenever the graph G is clear from the context. For vertex sets $X, Y \subseteq V(G)$, we define $N[X] = \bigcup_{v \in X} N[v]$, $N(X) = N[X] \setminus X$, $N_Y[X] = N[X] \cap Y$, and $N_Y(X) = N_Y[X] \setminus X$. The *degree* of a vertex v in a graph G is defined as $|N(v)|$, and we denote it by $\mathsf{deg}_G(v)$, or just $\mathsf{deg}(v)$ of the graph is clear from the context. For an integer $t \geq 1$, we denote by P_t (resp. I_t, K_t) the path (resp. edgeless graph, complete graph) on t vertices. For two integers $a, b \geq 1$, we denote by $K_{a,b}$ the bipartite graph with parts of sizes a and b.

A *clique* (resp. *independent set*) of a graph G is a set of vertices that are pairwise adjacent (resp. not adjacent). A graph property is *hereditary* if whenever it holds for a graph G, it holds for all its induced subgraphs as well. Note that the properties of being an edgeless or a complete graph or an independent set are hereditary. We denote by $\Delta(G)$ (resp. $\omega(G)$ the maximum vertex degree (resp. clique size) of a graph G.

A *vertex cover* of a graph G is a set of vertices containing at least one endpoint of every edge, and it is *minimal* if no proper subset of it is a vertex cover. The main problem we study in the paper is formally stated as follows. We state it as a decision problem, since most of our results consider its parameterization by the solution size k.

> MAXIMUM MINIMAL VERTEX COVER (MMVC)
> **Input:** A graph G and a positive integer k.
> **Question:** Does G contain a minimal vertex cover of size at least k?

The following observation has been already used in previous work [16, 50].

▶ **Observation 1.** *Let G be a graph. A set $X \subseteq V(G)$ is a minimal vertex cover of G if and only if X is a vertex cover of G and, for every vertex $v \in X$, $N(v) \nsubseteq X$.*

The next lemma provides a useful way to conclude that we are dealing with a yes-instance in our kernelization algorithms.

▶ **Lemma 2.** *Let G be a graph and let $S \subseteq V(G)$ be an independent set. There exists a minimal vertex cover of G containing $N(S)$.*

Proof. Note that, since S is an independent set, $V(G) \setminus S$ is a vertex cover of G. Hence, there exists a minimal vertex cover X of G such that $X \subseteq V(G) \setminus S$. We claim that $N(S) \subseteq X$. Suppose for the sake of contradiction that there exists a vertex $v \in N(S)$ such that $v \notin X$. Since v has a neighbor u in S and $S \cap X = \emptyset$, the edge $\{u, v\}$ would not be covered by X. ◀

Note that, in particular, Lemma 2 implies that if (G, k) is an instance of the MAXIMUM MINIMAL VERTEX COVER problem and $v \in V(G)$ is a vertex of degree at least k, then we can conclude that (G, k) is a yes-instance. This will allow us to assume, in our kernelization algorithms, that $\Delta(G) \leq k - 1$.

Parameterized complexity. We refer the reader to [21, 26] for basic background on parameterized complexity, and we recall here only some basic definitions used in this article. A *parameterized problem* is a language $L \subseteq \Sigma^* \times \mathbb{N}$. For an instance $I = (x, k) \in \Sigma^* \times \mathbb{N}$, k is called the *parameter*.

A parameterized problem is *fixed-parameter tractable* (FPT) if there exists an algorithm $\mathcal{A}$, a computable function f, and a constant c such that given an instance $I = (x, k)$, $\mathcal{A}$ (called an FPT *algorithm*) correctly decides whether $I \in L$ in time bounded by $f(k) \cdot |I|^c$. For instance, the Vertex Cover problem parameterized by the size of the solution is FPT.

For an instance (x, k) of a parameterized problem Q, a *kernelization algorithm* is an algorithm $\mathcal{A}$ that, in polynomial time, generates from (x, k) an equivalent instance (x', k') of Q such that $|x'| + k' \leq f(k)$, for some computable function $f : \mathbb{N} \to \mathbb{N}$, where $|x'|$ denotes the size of x'. If $f(k)$ is bounded from above by a polynomial of the parameter, we say that Q admits a *polynomial kernel*. In particular, if $f(k)$ is bounded by a linear (resp. quadratic) function, then we say that Q admits a *linear* (resp. *quadratic*) kernel.

A *polynomial parameter transformation*, abbreviated as PPT, is an algorithm that, given an instance (x, k) of a parameterized problem A, runs in time polynomial in $|x|$ and outputs an instance (x', k') of a parameterized problem B such that k' is bounded from above by a polynomial on k and (x, k) is positive if and only if (x', k') is positive. If a parameterized problem A does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly and there exists a PPT from A to a parameterized problem B, then B does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly either [21].

3 A general framework for ruling out certain polynomial kernels

In this section we introduce our generic framework to obtain lower bounds on the degree of a "certain type" of polynomial kernels, which we call lop-*kernels* (see Definition 8), for parameterized vertex-maximization problems. Namely, we prove the following theorem. Note that, when applying it to a concrete problem Π, the inapproximability of Π will rely on some complexity assumption, typically P $\neq$ NP.

▶ **Theorem 3.** *Let Π be a vertex-maximization problem whose decision version is in* NP*, and suppose that Π does not admit a polynomial-time approximation algorithm with ratio $\mathcal{O}(n^{r-\varepsilon})$ on n-vertex graphs for $r, \varepsilon \in (0, 1]$. Then Π parameterized by the solution size does not admit a* lop*-kernel with $\mathcal{O}(k^{\frac{1}{1-r}-\varepsilon'})$ vertices for $\varepsilon' = \frac{\varepsilon}{(1-r+\varepsilon)(1-r)}$ when $r \in (0, 1)$, or with $\mathcal{O}(k^{\frac{1}{\varepsilon}})$ vertices when $r = 1$.*

Boria et al. [16] proved that the Maximum Minimal Vertex Cover problem does not admit an $\mathcal{O}(n^{\frac{1}{2}-\varepsilon})$-approximation algorithm for any $\varepsilon > 0$, unless P = NP. Hence, by applying Theorem 3 with $r = \frac{1}{2}$ we obtain the following corollary.

▶ **Corollary 4.** Maximum Minimal Vertex Cover *parameterized by the solution size does not admit a* lop*-kernel with $\mathcal{O}(k^{2-\varepsilon})$ vertices for any $\varepsilon > 0$, unless* P = NP.

Dublois et al. [27] recently provided a cubic kernel for the Maximum Minimal Feedback Vertex Set problem (defined naturally) parameterized by the solution size, and proved that the problem does not admit an $\mathcal{O}(n^{\frac{2}{3}-\varepsilon})$-approximation algorithm for any $\varepsilon > 0$, unless P = NP. Hence, by applying Theorem 3 with $r = \frac{2}{3}$ we obtain the following corollary, which states that the cubic kernel of Dublois et al. [27] is essentially optimal.

▶ **Corollary 5.** Maximum Minimal Feedback Vertex Set *parameterized by the solution size does not admit a* lop*-kernel with $\mathcal{O}(k^{3-\varepsilon})$ vertices for any $\varepsilon > 0$, unless* P = NP.

In the remainder of this section we prove Theorem 3. In order to do so, we present simple self-contained ad-hoc arguments relating the kernel size to the approximability of the considered problem.

We define lop-rules for an arbitrary vertex-maximization problem Π such that the input is a graph G, the output is a subset $S \subseteq V(G)$ satisfying some conditions, and the goal is to maximize $|S|$. Given a graph G and an integer k, we say that (G, k) is a yes-*instance* of Π if $\mathsf{opt}_\Pi(G) \geq k$, where $\mathsf{opt}_\Pi(G)$ denotes the maximum size of a solution of Π in G.

▶ **Definition 6.** *A* large optimal preserving *reduction rule, or* lop-rule *for short, for a vertex-maximization problem* Π*, is a polynomial-time algorithm* R *that, given a pair* (G, k)*, where* G *is a graph and* k *is a positive integer, computes another pair* (G', k') *with* $0 \leq k' \leq k$ *such that*

1. *if* (G, k) *is a* no-*instance of* Π*, then* (G', k') *is a* no-*instance of* Π*, and*
2. *if* (G, k) *is a* yes-*instance of* Π*, then* $\mathsf{opt}_\Pi(G') \geq \mathsf{opt}_\Pi(G) - (k - k')$*, implying that* (G', k') *is a* yes-*instance of* Π.

Note that Property 2 in Definition 6 is stronger than the implication "if (G, k) is a yes-instance of Π, then (G', k') is a yes-instance of Π", which would yield a classical kernelization algorithm. Indeed, when we consider how this latter implication is generally proved in safeness proofs of classical kernels, one of the following scenarios often occur:

(a) For every solution S in G there exists a solution S' in G' with $|S'| \geq |S| - (k - k')$.
(b) If there exists a solution S in G with $|S| \geq k$, then there exists a solution S' in G' with $|S'| \geq |S| - (k - k')$.
(c) If there exists a solution S in G with $|S| \geq k$, then there exists a solution S' in G' with $|S'| \geq k'$.

In Case (a), the rule preserves all optimal solutions, and it implies that $\mathsf{opt}_\Pi(G') \geq \mathsf{opt}_\Pi(G) - (k - k')$. In Case (b), the rule preserves only large optimal solutions, and it implies that if $\mathsf{opt}_\Pi(G) \geq k$, then $\mathsf{opt}_\Pi(G') \geq \mathsf{opt}_\Pi(G) - (k - k')$, implying Property 2 above; note that if $\mathsf{opt}_\Pi(G) < k$, then $\mathsf{opt}_\Pi(G')$ and $\mathsf{opt}_\Pi(G)$ are not necessarily related. Case (c) corresponds to the weaker and classical implication "if (G, k) is a yes-instance of Π, then (G', k') is a yes-instance of Π".

The following observation is an immediate consequence of the definition of a lop-rule.

▶ **Observation 7.** lop-*rules can be composed. Formally, consider two* lop-*rules* R_1 *and* R_2*. Then, the rule* R *that, given a instance* (G, k)*, returns* $R_2(R_1(G, k))$ *is a* lop-*rule.*

A typical example of a lop-rule is when we can identify a "dominant" set of vertices that can be safely included into a solution. More precisely, consider a rule that finds a subset $T \subseteq V(G)$ and a graph G' such that there exists an optimal solution $S^\star$ in G such that $S^\star = T \cup S'$, where S' is a solution in G', and for every solution S' in G', $S' \cup T$ is a solution in G. Such a rule is a lop-rule, as we even fall into Case (a) described above.

Even if we are not aware of known reduction rules for vertex-maximization problems that are *not* lop-rules, we can artificially devise such an example. For instance, for the MMVC problem, given an instance (G, k), if there is a vertex that has more than k neighbors of degree one, we can safely delete all but any k of them to obtain a reduced graph G', and leave k unchanged. Note that this rule falls into Case (c) above, since by Lemma 2 both G and G' are yes-instances of MMVC, but it does not satisfy Property 2 in Definition 6, since $\mathsf{mmvc}(G)$ may be arbitrarily larger than $\mathsf{mmvc}(G')$.

If we defined a lop-kernel as an algorithm consisting only of lop-rules, we would exclude from being a lop kernel, for instance, a rule that detects a yes-instance as in the above paragraph. This justifies the next definition, where we allow lop-kernels to decide instances.

▶ **Definition 8.** *Let* Π *be a vertex-maximization problem and let* $s : \mathbb{N} \to \mathbb{N}$ *be a computable function. A* lop-kernel of size s *for* Π *parameterized by the solution size is a polynomial-time algorithm that takes as input an instance* (G, k)*, produces a reduced instance* (G', k') *by applying a (possibly empty) sequence of* lop*-rules to* (G, k)*, and either*

- *determines that* (G', k') *is a* yes*-instance or a* no*-instance, or*
- *outputs* (G', k') *with* $|V(G')| \leq s(k)$*.*

Note that, as lop-rules provide equivalent instances, if the lop-kernel falls into the first case (where it correctly decides (G', k')), then it also correctly decides (G, k). On the other hand, if it falls into the second case (where it outputs (G', k')), and the kernel has size $\mathcal{O}(k^c)$ for some constant $c \geq 1$, then Property 2 implies that $\mathsf{opt}_\Pi(G) \leq \mathsf{opt}_\Pi(G') + (k - k') \leq |V(G')| + k = \mathcal{O}(k^c)$. Hence, a lop-kernel of size $\mathcal{O}(k^c)$ yields a polynomial-time algorithm certifying either that $\mathsf{opt}_\Pi(G) \geq k$ (when it decides that (G, k) is a yes-instance), or that $\mathsf{opt}_\Pi(G) = \mathcal{O}(k^c)$ (when it decides that (G, k) is a no-instance, or outputs (G', k')).

Our next objective is to use such a polynomial-time algorithm in order to obtain an approximation algorithm for the considered problem. For this, we need the following definition, which is inspired by a similar notion introduced by Hochbaum and Shmoys [41], and referred to as *f-relaxed decision procedure* in [48]. Note that the definition in [41] is for a minimization problem, and their algorithm has either to certify that $\mathsf{opt}_\Pi(G) > k$, or to produce a solution (which is not required here) of size at most $f(k)$.

▶ **Definition 9.** *Let* Π *be a vertex-maximization problem and let* $f : \mathbb{N} \to \mathbb{N}$ *be a function. An* f-dual-approximation algorithm for Π *is a polynomial-time algorithm that, given a graph* G *and a positive integer* k*, concludes one of the following:*

- $\mathsf{opt}_\Pi(G) \geq k$*.*
- $\mathsf{opt}_\Pi(G) < f(k)$*.*

In the next lemma we prove that a lop-kernel of size s yields an f-dual-approximation algorithm (where f depends on s), which in turn yields a classical approximation algorithm whose ratio depends on s. For the latter implication, proved in Lemma 11, we restrict ourselves to functions s that are polynomial, since our goal is to obtain lower bounds on the degree of polynomial kernels.

▶ **Lemma 10.** *Let* Π *be a vertex-maximization problem and let* $s : \mathbb{N} \to \mathbb{N}$ *be a computable function. If* Π *parameterized by the solution size* k *admits a* lop*-kernel of size* $s(k)$*, then* Π *admits an* f*-dual-approximation algorithm with* $f(k) := s(k) + k + 1$*.*

Proof. Let A be a lop-kernel of size $s(k)$ for Π parameterized by the solution size. Given an instance (G, k), let (G', k') be the instance computed by A (using only lop-rules). Observe first that, according to Observation 7, we have that the sequence of lop-rules involved in the reduction from (G, k) to (G', k') is equivalent to a single lop-rule that transforms (G, k) into (G', k'). If A concludes that (G', k') is a yes-instance, then as a lop-rule provides an equivalent instance, we conclude that (G, k) is a yes-instance, and fall into the first item of Definition 9.

Otherwise, we claim that $\mathsf{opt}_\Pi(G) < f(k)$. If A concludes that (G', k') is a no-instance, then we conclude that (G, k) is a no-instance. This implies $\mathsf{opt}_\Pi(G) < k$, and we are done because $k \leq f(k)$ for every $k \geq 0$. It remains to consider the case where A outputs (G', k'). If $\mathsf{opt}_\Pi(G) < k$ then again we are done. Otherwise, as $\mathsf{opt}_\Pi(G) \geq k$, Property 2 in Definition 6 implies that $\mathsf{opt}_\Pi(G) \leq \mathsf{opt}_\Pi(G') + (k - k') \leq |V(G')| + k < s(k) + k + 1 = f(k)$. ◀

In the proof of the next lemma we need the hypothesis that decision version of the considered problem Π belongs to NP, to be able to verify in polynomial time if a vertex subset is a solution. Due to this, we also need this hypothesis in Theorem 3.

▶ **Lemma 11.** *Let Π be a vertex-maximization problem whose decision version is in* NP*, $c > 1$ be a real number, and $f : \mathbb{N} \to \mathbb{N}$ be a computable function with $f(k) = \mathcal{O}(k^c)$. If Π admits an f-dual-approximation algorithm, then Π admits a polynomial-time approximation algorithm with ratio $\mathcal{O}(n^{\frac{c-1}{c}})$ on n-vertex graphs.*

Proof. Let A be an f-dual-approximation algorithm for Π. We proceed to construct a polynomial-time approximation algorithm for Π with the claimed ratio[1]. Given an n-vertex graph G, let k_0 be the largest positive integer k such that algorithm A returns that $\mathsf{opt}_\Pi(G) \geq k$. Note that k_0 can be found in polynomial time by performing at most n calls to algorithm A. If $k_0 = 0$, we have that $\mathsf{opt}_\Pi(G) < f(0) = \mathcal{O}(1)$, and since the decision version of Π is in NP, we can find an optimal solution in polynomial time by verifying all vertex subsets of size at most $f(0)$. Otherwise, that is, when $k_0 \geq 1$, our approximation algorithm returns k_0. Let us prove that it provides the claimed approximation ratio.

We distinguish two cases depending on the value of k_0. Suppose first that $k_0 \geq n^{1/c}$. Since $\mathsf{opt}_\Pi(G) \leq n$, in this case we get that

$$\frac{\mathsf{opt}_\Pi(G)}{k_0} \leq \frac{n}{n^{1/c}} = n^{\frac{c-1}{c}}. \tag{1}$$

Otherwise, it holds that $k_0 < n^{1/c}$. By the definition of k_0 we have that $\mathsf{opt}_\Pi(G) < f(k_0 + 1) = \mathcal{O}((k_0 + 1)^c) = \mathcal{O}((k_0)^c)$. Thus, in this case we get that

$$\frac{\mathsf{opt}_\Pi(G)}{k_0} = \frac{\mathcal{O}((k_0)^c)}{k_0} = \mathcal{O}\left((k_0)^{c-1}\right) = \mathcal{O}\left(n^{\frac{c-1}{c}}\right). \tag{2}$$

Since in both cases we have a ratio of $\mathcal{O}(n^{\frac{c-1}{c}})$, the lemma follows. ◀

We finally have all the ingredients to prove Theorem 3.

Proof of Theorem 3. We prove that Π parameterized by the solution size does not admit a lop-kernel with $\mathcal{O}(k^{\frac{1}{1-r+\varepsilon}})$ vertices. Once this is proved, the theorem follows since $\frac{1}{1-r+\varepsilon}$ is equal to $\frac{1}{\varepsilon}$ when $r = 1$ and $\frac{1}{1-r} - \varepsilon'$ for $\varepsilon' = \frac{\varepsilon}{(1-r+\varepsilon)(1-r)}$ when $r \in (0, 1)$.

Assume to the contrary that Π parameterized by the solution size k admits a lop-kernel with $\mathcal{O}(k^{\frac{1}{1-r+\varepsilon}})$ vertices. By Lemma 10 it follows that Π admits an f-dual-approximation algorithm with $f(k) = \mathcal{O}(k^{\frac{1}{1-r+\varepsilon}}) + k + 1 = \mathcal{O}(k^{\frac{1}{1-r+\varepsilon}})$. We use this to get a contradiction by applying Lemma 11 for $c = \frac{1}{1-r+\varepsilon}$ and obtain that Π admits a polynomial-time approximation algorithm with ratio $\mathcal{O}(n^{r-\varepsilon})$ on n-vertex graphs. ◀

4 Our framework for vertex-minimization problems

In this section we provide the definitions of lop-kernel for vertex-minimization problems and present the corresponding result, Theorem 13, which is the translation of Theorem 3 to vertex-minimization problems. All the details can be found in the full version [3].

[1] We consider here the problem of computing an approximation of the optimal value, and not constructing the corresponding solution. This is not restrictive, as the type of inapproximability results that allow the application of Theorem 3, such as [16, 27], also apply to this case.

We consider an arbitrary vertex-minimization problem Π such that the input is a graph G, the output is a subset $S \subseteq V(G)$ satisfying some conditions, and the goal is to minimize $|S|$. Given a graph G and an integer k, we say that (G, k) is a yes-*instance* of Π if $\mathsf{opt}_\Pi(G) \leq k$, where $\mathsf{opt}_\Pi(G)$ denotes the minimum size of a solution of Π in G. In order to state our general result, we first need to define lop-rules and lop-kernels for vertex-minimization problems.

▶ **Definition 12.** *A* large optimal preserving *reduction rule, or* lop-rule *for short, for a vertex-minimization problem* Π*, is a polynomial-time algorithm* R *that, given a pair* (G, k)*, where* G *is a graph and* k *is a positive integer, computes another pair* (G', k') *with* $0 \leq k' \leq k$ *such that*

1. *if* (G, k) *is a* yes-*instance of* Π*, then* (G', k') *is a* yes-*instance of* Π*, and*
2. *if* (G, k) *is a* no-*instance of* Π*, then* $\mathsf{opt}_\Pi(G') \geq \mathsf{opt}_\Pi(G) - (k - k')$*, implying that* (G', k') *is a* no-*instance of* Π.

Recall (cf. Definition 6) that a lop-rule for a vertex-maximization problem can be seen as a classical kernel whose property "if (G, k) is a yes-instance, then (G', k') is a yes-instance" is strengthened, whereas Property 2 in Definition 12 is a reinforcement of the classical implication "if (G, k) is a no-instance of Π, then (G', k') is a no-instance of Π". This apparent lack of symmetry is due to technical reasons that make it possible to prove Theorem 13. However, both versions look more similar if we rewrite them in the following way. Namely, the two properties of Definition 6 can be rewritten as

1. if $\mathsf{opt}_\Pi(G) < k$, then $\mathsf{opt}_\Pi(G') < k'$, and
2. if $\mathsf{opt}_\Pi(G) \geq k$, then $\mathsf{opt}_\Pi(G') \geq \mathsf{opt}_\Pi(G) - (k - k')$, implying that $\mathsf{opt}_\Pi(G') \geq k'$,

and the two properties of Definition 12 can be rewritten as

1. if $\mathsf{opt}_\Pi(G) \leq k$, then $\mathsf{opt}_\Pi(G') \leq k'$, and
2. if $\mathsf{opt}_\Pi(G) > k$, then $\mathsf{opt}_\Pi(G') \geq \mathsf{opt}_\Pi(G) - (k - k')$, implying that $\mathsf{opt}_\Pi(G') > k'$.

Indeed, observe that the above conditions are exactly the same in both definitions, up to strict inequalities. Note also that, in both cases, these rules preserve "large solutions" in G into G', and this is why we call both of them "lop-rules".

Once we have defined lop-rules for vertex-minimization problems, the definition of lop-*kernel for a vertex-minimization problem* is the same as for vertex-maximization problems, that is, the same as Definition 8 by just replacing "vertex-maximization" with "vertex-minimization".

Note that, as lop-rules provide equivalent instances, if the lop-kernel falls into the first case (where it correctly decides (G', k')), then it also correctly decides (G, k). On the other hand, if it falls into the second case (where it outputs (G', k')), and the kernel has size $\mathcal{O}(k^c)$ for some constant $c \geq 1$, then Property 2 implies that $\mathsf{opt}_\Pi(G) \leq \mathsf{opt}_\Pi(G') + (k - k') \leq |V(G')| + k = \mathcal{O}(k^c)$. Hence, a lop-kernel of size $\mathcal{O}(k^c)$ yields a polynomial-time algorithm certifying either that $\mathsf{opt}_\Pi(G) > k$ (when it decides that (G, k) is a no-instance), or that $\mathsf{opt}_\Pi(G) = \mathcal{O}(k^c)$ (when it either decides that (G, k) is a yes-instance, or outputs (G', k')).

As in the case of vertex-maximization problems, we can use such a polynomial-time algorithm in order to obtain an approximation algorithm for the considered problem, and this is the main idea behind the proof of Theorem 13, which can be found in the full version [3].

▶ **Theorem 13.** *Let* Π *be a vertex-minimization problem whose decision version in* NP*, and suppose that* Π *does not admit a polynomial-time approximation algorithm with ratio* $\mathcal{O}(n^{r-\varepsilon})$ *on* n*-vertex graphs for* $r, \varepsilon \in (0, 1]$*. Then* Π *parameterized by the solution size does not admit a* lop-*kernel with* $\mathcal{O}(k^{\frac{1}{1-r} - \varepsilon'})$ *vertices for* $\varepsilon' = \frac{\varepsilon}{(1-r+\varepsilon)(1-r)}$ *when* $r \in (0, 1)$*, or with* $\mathcal{O}(k^{\frac{1}{\varepsilon}})$ *vertices when* $r = 1$.

5 An attempt to obtain a linear kernel for MMVC

In this section we briefly explain the flaw in the linear kernel for MMVC claimed by Fernau [30, Corollary 4.25], and that is based on joint unpublished work with Dehne, Fellows, Prieto, and Rosamond. The kernelization algorithm is a small modification of a linear kernel for the NONBLOCKER SET problem presented by Ore [46]. A set of vertices S of a graph G is a *nonblocker* if its complement is a dominating set of G, that is, for every $u \in S$ there exists $v \notin S$ with $\{u, v\} \in E(G)$. In the NONBLOCKER SET problem, we are given a graph G and an integer parameter k, and the goal is to decide whether G contains a nonblocker of size at least k. Suppose for simplicity that G is connected. The idea is to consider an arbitrary spanning tree T of G, root it arbitrarily at a vertex r, and partition $V(G) = V_0 \uplus V_1$ such that the vertices in V_0 (resp. V_1) are within even (resp. odd) distance from r in T. By construction, each of V_0 and V_1 is a nonblocker in G, so if one of them has size at least k, we can answer "yes", and otherwise $|V(G)| \leq 2k$ and we are done.

Back to MMVC, it is observed in [30, Reduction rule 24] that a simple reduction rule allows to assume that no connected component of G is a clique (in particular, an isolated vertex). Assume again for simplicity that G is connected. It is then claimed in [30] that, using the same algorithm as for NONBLOCKER SET, the largest of V_0 and V_1, say V_0, can be always completed into a minimal vertex cover of G, which would immediately yield a kernel of size at most $2k$ for MMVC. Unfortunately, this claim is not true: when adding new vertices to V_0 in order to make it a vertex cover of G, we may lose the minimality property, and some vertices may need to be removed. For instance, let G be the graph obtained from a triangle on vertices u, v, w by adding $p \geq 2$ pendant vertices to each of u, v, and w. Let T be the spanning tree obtained from G by removing the edge $\{v, w\}$, and root T at vertex u. Then $|V_0| = 1 + 2p$ and $|V_1| = 2 + p$, so $|V_0| > |V_1|$, and note that the edge $\{v, w\}$ is the only edge of G not covered by V_0. But adding either of v or w to V_0, say v, results in a non-minimal vertex cover of G, and therefore the p pendant vertices adjacent to v have to be removed from V_0, which yields a set of size $2 + p < \frac{|V(G)|}{2} = \frac{3+3p}{2}$, where we have used that $p \geq 2$. In fact, deciding whether a set $S \subseteq V(G)$ can the extended to a minimal vertex cover of G is an NP-complete problem [17].

6 A subquadratic kernel for MMVC on bull-free graphs

In this section we present a subquadratic kernel for MAXIMUM MINIMAL VERTEX COVER restricted to bull-free graphs when the parameter is the solution size k. Subquadratic kernels on other graph classes, as well as other positive results, can be found in the full version [3], namely on K_t-free graphs, t-bull-free graphs (that generalize bull-free graphs), paw-free graphs, $K_{1,t}$-free graphs, graphs with bounded chromatic number, and graphs with bounded cliquewidth.

For a constant $\delta > 0$, a graph H is said to satisfy the *Erdős-Hajnal property with constant δ* if every H-free graph G on n vertices contains either a clique or an independent set of size n^δ. The (still open) Erdős-Hajnal conjecture [29] states that every graph H satisfies the Erdős-Hajnal property. As reported by Chudnovsky [19], the Erdős-Hajnal conjecture has been verified for only a small number of graphs, namely all graphs on at most four vertices, the *bull* (i.e., the graph obtained by adding a pendant vertex to two different vertices of a triangle), the complete graphs, and every graph that can be constructed from them using the so-called *substitution operation* [2], which we define later.

Since our goal is to use the Erdős-Hajnal property in order to obtain kernels for Maximum Minimal Vertex Cover, we need an algorithmic version of it. As defined by Bonnet et al. [15], for a constant $\delta > 0$, a graph H is said to satisfy the *constructive Erdős-Hajnal property with constant* δ if there exists an algorithm that takes as input an H-free graph G on n vertices, and outputs in polynomial-time a clique or an independent set of G of size at least n^δ. Fortunately for our purposes, all the graphs H shown to satisfy the Erdős-Hajnal property so far, also satisfy its constructive version [15].

In the following lemma, we show that, if H is a graph satisfying the constructive Erdős-Hajnal property, then the vertex set of an H-free graph can be partitioned in polynomial time into "few" cliques or independent sets. This partition will then be used to obtain subquadratic kernels on H-free graphs for several graphs H.

▶ **Lemma 14.** *Let H be a graph satisfying the constructive Erdős-Hajnal property with constant δ. The vertex set of any H-free graph G on n vertices can be partitioned in polynomial time into a collection of cliques $\mathcal{C}$ and a collection of independent sets $\mathcal{I}$ such that $|\mathcal{C}| + |\mathcal{I}| \leq \left(\frac{1}{2^{(1-\delta)}-1}\right) \cdot n^{1-\delta}$.*

Proof. Let G be an H-free graph on n vertices. We initialize $X_0 = V(G), \mathcal{C} = \mathcal{I} = \emptyset$, and we run the following procedure as far as $|X_0| \geq 1$:

> Find in polynomial time a clique or an independent set Y in $G[X_0]$ with $|Y| \geq |X_0|^\delta$. Note that this is possible since $G[X_0]$ is an H-free graph for any $X_0 \subseteq V(G)$. Add Y to $\mathcal{C}$ or to $\mathcal{I}$ depending on whether Y is a clique or an independent set, respectively (if $|Y| = 1$, choose $\mathcal{C}$ or $\mathcal{I}$ arbitrarily). Update $X_0 \leftarrow X_0 \setminus Y$.

Clearly, the above algorithm terminates in polynomial time. It remains to bound $|\mathcal{C}| + |\mathcal{I}|$, which is equal to the number of iterations of the algorithm. To this end, for a positive integer i, we say that an iteration belongs to *step i* of the algorithm if the current set X_0 at the start of the iteration satisfies $\frac{n}{2^i} < |X_0| \leq \frac{n}{2^{i-1}}$. We denote by t_i the number of iterations of the algorithm within step i. By definition, $|\mathcal{C}| + |\mathcal{I}| = \sum_{i=1}^{\infty} t_i$. Let Y be a clique or an independent set found by the algorithm within step i. Since the current set X_0 satisfies $|X_0| > \frac{n}{2^i}$, we have that $|Y| > \left(\frac{n}{2^i}\right)^\delta$. And since the sum of the sizes of the sets found before the last iteration of step i is at most $\frac{n}{2^i}$, it follows that $t_i \leq \left(\frac{n}{2^i}\right)^{1-\delta}$. Note that, in particular, $t_i = 0$ for $i > \lceil \log n \rceil$. Therefore, we conclude that

$$|\mathcal{C}| + |\mathcal{I}| \;=\; \sum_{i=1}^{\infty} t_i \;\leq\; \sum_{i=1}^{\infty} \left(\frac{n}{2^i}\right)^{1-\delta} \;=\; n^{1-\delta} \cdot \sum_{i=1}^{\infty} \left(\frac{1}{2^{1-\delta}}\right)^i \;=\; n^{1-\delta} \cdot \left(\frac{1}{2^{(1-\delta)}-1}\right),$$

and the lemma follows. ◀

We are now ready to present the subquadratic kernel on bull-free graphs. Note that, since bipartite graphs are bull-free, MMVC restricted to bull-free graphs is NP-hard by [12] (or by Theorem 17). In the kernels presented in this section and in the full version, since we can easily obtain explicit constants, we decided not to use the big-O notation.

▶ **Theorem 15.** Maximum Minimal Vertex Cover *parameterized by k restricted to bull-free graphs admits a kernel with at most $c(k-1)^{7/4} + k - 1$ vertices, where $c = \frac{2}{2^{\frac{3}{4}}-1} < 3$.*

Proof. Let (G, k) be an instance of the Maximum Minimal Vertex Cover problem, where G is a bull-free graph. Recall that by Lemma 2 we can assume that the maximum degree of G is at most $k-1$. We start by finding greedily, starting from $V(G)$, a minimal vertex cover X of G. Note that X can be easily found in polynomial time by Observation 1. If $|X| \geq k$, we conclude that (G,k) is a yes-instance, so we can assume that $|X| \leq k-1$. Let $S = V(G) \setminus X$ and note that S is an independent set.

Since the bull satisfies the constructive Erdős-Hajnal property with constant $\delta = \frac{1}{4}$ [15,20], we can apply Lemma 14 to the bull-free graph $G[X]$ and obtain in polynomial time a partition of X into a collection of cliques $\mathcal{C}$ and a collection of independent sets $\mathcal{I}$ such that $|\mathcal{C}| + |\mathcal{I}| \leq d \cdot |X|^{3/4} \leq d \cdot (k-1)^{3/4}$, where $d = \frac{1}{2^{\frac{3}{4}}-1} < 1.47$. Since we can assume that G has no isolated vertices, as they can be safely removed without affecting the type of the instance, it follows that

$$S = \bigcup_{C \in \mathcal{C}} N_S(C) \cup \bigcup_{I \in \mathcal{I}} N_S(I). \tag{3}$$

Hence, our objective is to bound $|N_S(Y)|$ for every $Y \in \mathcal{C} \cup \mathcal{I}$. Suppose first that $I \in \mathcal{I}$ is an independent set. From Lemma 2, if $|N_S(I)| \geq k$ we can conclude that (G,k) is a yes-instance, so we can assume henceforth that

$$\text{for every independent set } I \in \mathcal{I}, \text{ it holds } \ |N_S(I)| \leq k-1. \tag{4}$$

Suppose now that $C \in \mathcal{C}$ is a clique. We partition $N_S(C) = S_C^1 \uplus S_C^2$ as follows. Let S_C^1 be an inclusion-wise maximal set of vertices in $N_S(C)$ such that for any two (not necessarily distinct) vertices $x, y \in S_C^1$, $|N_C(x) \cup N_C(y)| \leq |C| - 1$. That is, S_C^1 is a maximal set in $N_S(C)$ such that the neighborhoods of its vertices pairwise do *not* cover the whole clique C. We let $S_C^2 = N_S(C) \setminus S_C^1$. The following is the crucial property of the set S_C^1.

▷ Claim 16. The vertices in S_C^1 can be ordered $x_1, \ldots, x_p$ so that $N_C(x_i) \subseteq N_C(x_j)$ if $i \leq j$.

Proof. In order to prove the claim, it is sufficient to prove that, for any two vertices $x, y \in S_C^1$, either $N_C(x) \subseteq N_C(y)$ or $N_C(y) \subseteq N_C(x)$. Suppose for the sake of contradiction that there exist two vertices $u \in N_C(x) \setminus N_C(y)$ and $v \in N_C(y) \setminus N_C(x)$. By definition of the set S_C^1, there exists a vertex $w \in C \setminus (N_C(x) \cup N_C(y))$. But then the vertices x, y, u, v, w induce a bull as illustrated in Figure 1, contradicting the hypothesis that G is bull-free. ◁

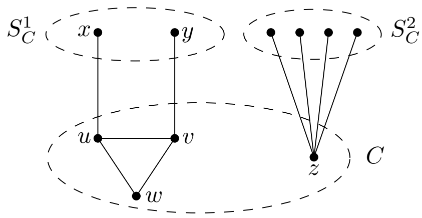

Figure 1 Configuration considered in the proof of Claim 16 and a vertex $z \in \bigcap_{x \in S_C^2} N_C(x)$.

Claim 16 implies in particular that, unless $S_C^1 = \emptyset$, there exists a vertex $u \in \bigcap_{x \in S_C^1} N_C(x)$. Since u has degree at most $k-1$ in G, and each vertex $x \in S_C^1$ is adjacent to u, it follows that $|S_C^1| \leq k-1$. Let us now focus on the set S_C^2. The definition of the set S_C^1 together with Claim 16 imply that there exists a vertex $z \in C \setminus \bigcup_{y \in S_C^1} N_C(y)$. Consider now an arbitrary vertex $x \in S_C^2$. Since x could not be added to S_C^1, there exists a vertex $y \in S_C^1$ such that $N_C(x) \cup N_C(y) = C$. But since $z \in C \setminus \bigcup_{y \in S_C^1} N_C(y)$, necessarily $z \in N_C(x)$. It follows that $z \in \bigcap_{x \in S_C^2} N_C(x)$ (see Figure 1). Using again the fact that z has degree at most $k-1$ in G, we obtain that $|S_C^2| \leq k-1$. Summarizing, we have that

$$\text{for every clique } C \in \mathcal{C}, \text{ it holds } \ |N_S(C)| = |S_C^1| + |S_C^2| \leq 2(k-1). \tag{5}$$

Putting all pieces together, Equations (3), (4), and (5) and the fact that $|X| \leq k-1$ and $|\mathcal{C}|+|\mathcal{I}| \leq d \cdot |X|^{3/4}$ imply that, unless we have already concluded that (G,k) is a yes-instance,

$$\begin{aligned}|V(G)| = |X| + |S| \ &= \ |X| + |\bigcup_{C\in\mathcal{C}} N_S(C)| + |\bigcup_{I\in\mathcal{I}} N_S(I)| \\ &\leq |X| + (|\mathcal{C}|+|\mathcal{I}|) \cdot \max_{Y\in\mathcal{C}\cup\mathcal{I}} |N_S(Y)| \ \leq \ k-1+d\cdot(k-1)^{3/4}\cdot 2(k-1) \\ &= 2d\cdot(k-1)^{7/4}+k-1, \qquad \text{and the theorem follows.}\end{aligned}$$ ◀

7 Ruling out polynomial kernels for MMVC for smaller parameters

In this section we rule out, assuming that NP $\not\subseteq$ coNP/poly, the existence of polynomial kernels for MMVC parameterized by the size of minimum vertex cover of the input graph. As mentioned in the introduction, the reduction given in Theorem 17 also provides an alternative proof of the NP-completeness of MMVC on bipartite graphs, which also follows from [12]. We note that the existing NP-hardness reductions for MMVC, such as the one in [12], do *not* seem to be easily modifiable so to yield the non-existence of polynomial kernels.

▶ **Theorem 17.** *The* Maximum Minimal Vertex Cover *problem parameterized by the size of a minimum vertex cover (or of a maximum matching) of the input graph does not admit a polynomial kernel unless* NP $\subseteq$ coNP/poly*, even restricted to bipartite graphs.*

Proof. We present a PPT from Monotone Sat parameterized by the number of variables, which is also an NP-completeness reduction. The Monotone Sat problem is the restriction of the Sat problem to formulas in which the literals in each clause are either all positive or all negative. This problem is well-known to be NP-complete [33], and it is easy to see that, when parameterized by the number of variables, it does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly. Indeed, Fortnow and Santhanam [32] proved that the Sat problem parameterized by the number of variables does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly, and the classical reduction from Sat to Monotone Sat that replaces each variable with a "positive" and a "negative" variable and adds extra clauses appropriately [33] is in fact a PPT when the parameter is the number of variables.

Given an instance ϕ of Monotone Sat, where the formula ϕ contains n variables and m clauses, we construct in polynomial time an instance (G,k) of Maximum Minimal Vertex Cover as follows. For each variable x_i of ϕ, $i \in [n]$, we add to G four vertices $\ell_i, x_i^+, x_i^-, r_i$ and three edges $\{\ell_i, x_i^+\}, \{x_i^+, x_i^-\}, \{x_i^-, r_i\}$, hence inducing a P_4. We call the vertex x_i^+ (resp. x_i^-) a *positive* (resp. a *negative*) vertex of G. For each clause C_j of ϕ, $j \in [m]$, we add to G a vertex c_j, which we connect to the positive or negative vertices corresponding to the literals contained in C_j. This concludes the construction of G, which is illustrated in Figure 2(a). Note that, since ϕ is a monotone formula, G is a bipartite graph. Note also that the set of vertices $\{x_i^+, x_i^- \mid i \in [n]\}$ is a minimum vertex cover of G of size $2n$, and that the set of edges $\{\{\ell_i, x_i^+\}, \{x_i^-, r_i\} \mid i \in [n]\}$ is a maximum matching of G of size $2n$. We claim that ϕ is satisfiable if and only if G contains a minimal vertex cover of size $k := 2n+m$.

Suppose first that ϕ is satisfiable, and let σ be an assignment of the variables that satisfies all the clauses in ϕ. We proceed to define a minimal vertex cover X of G of size k. First, add to X all the clause vertices $\{c_j \mid j \in [m]\}$. For every $i \in [n]$, if $\sigma(x_i) =$ true (resp. $\sigma(x_i) =$ false), add to X vertices x_i^- and ℓ_i (resp. x_i^+ and r_i). See Figure 2(b) for an illustration, where the set X is shown with larger red vertices. Clearly, X is a vertex cover of G. To see that it is minimal, by Observation 1 it is enough to verify that, for every vertex $v \in X$, $N[v] \not\subseteq X$. This condition holds easily for all vertices in X that are in the P_4's,

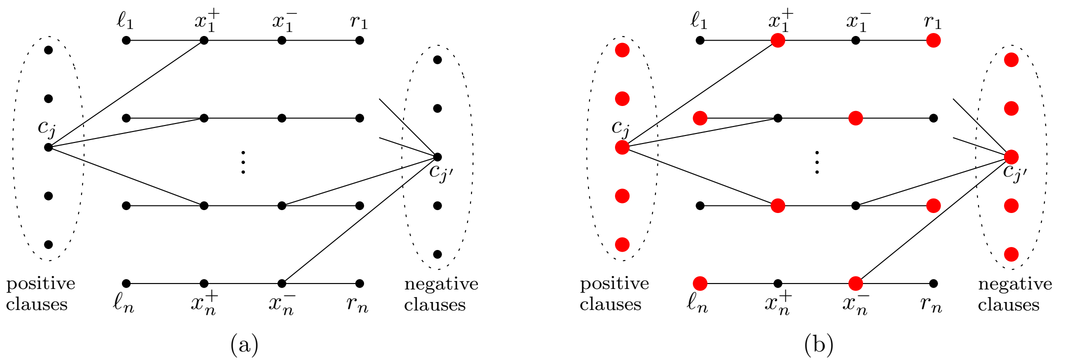

Figure 2 (a) Illustration of the graph G built from the formula ϕ in the proof of Theorem 17. (b) A minimal vertex cover X of G is shown with larger red vertices.

since for each P_4 its vertices in X are not adjacent. Let c_j be a clause vertex. Since σ is a satisfying assignment of the variables, there exists a variable x_i such that if $\sigma(x_i) = \mathsf{true}$ (resp. $\sigma(x_i) = \mathsf{false}$) then $x_i \in C_j$ (resp. $\bar{x}_i \in C_j$). By definition of X, if $\sigma(x_i) = \mathsf{true}$ (resp. $\sigma(x_i) = \mathsf{false}$) then $x_i^+ \notin X$ (resp. $x_i^- \notin X$), and by construction of G we have that $x_i^+ \in N(c_j)$ (resp. $x_i^- \in N(c_j)$), so in both cases $N[c_j] \nsubseteq X$.

Conversely, suppose that G contains a minimal vertex cover X of size k, and we proceed to define a variable assignment σ as follows. For $i \in [n]$, as $\{x_i^+, x_i^-\} \in E(G)$ we have that X contains one or two vertices in the set $\{x_i^+, x_i^-\}$. If $x_i^+ \notin X$ (resp. $x_i^- \notin X$) we set $\sigma(x_i) = \mathsf{true}$ (resp. $\sigma(x_i) = \mathsf{false}$), and if both x_i^+ and x_i^- belong to X we set $\sigma(x_i)$ to true or to false arbitrarily. We claim that σ satisfies all the clauses in ϕ. For $i \in [n]$, let P^i be the P_4 of G induced by the vertices $\ell_i, x_i^+, x_i^-, r_i$. Since X is a vertex cover, clearly $|X \cap V(P^i)| \geq 2$. We claim that $|X \cap V(P^i)| = 2$. Indeed, if $|X \cap V(P^i)| \geq 3$, then $\{\ell_i, x_i^+\} \subseteq X$ or $\{x_i^-, r_i\} \subseteq X$ (or both). But then $N[\ell_i] \subseteq X$ or $N[r_i] \subseteq X$ (or both), contradicting Observation 1. Thus, $|X \cap V(P^i)| = 2$, which implies that $|X \cap \bigcup_{i \in [n]} V(P^i)| = 2n$, hence necessarily X contains the whole set $\{c_j \mid j \in [m]\}$ of clause vertices. Consider an arbitrary clause vertex c_j. Since X is minimal and $c_j \in X$, by Observation 1 there exists a neighbor of c_j in G that is *not* in X, and by definition of σ it follows that the literal corresponding to that neighbor of c_j satisfies clause C_j. Thus, σ is a satisfying assignment and the proof is complete.

Finally, note that the above reduction is also an NP-completeness reduction from Monotone Sat to Maximum Minimal Vertex Cover on bipartite graphs. ◀

8 Conclusions and further research

We presented a framework to obtain lower bounds on the degrees of certain types of polynomial kernels, which we called lop-kernels, for vertex-maximization and vertex-minimization problems. Note that the classical kernels for Vertex Cover such as those using the high-degree rule, the crown decomposition rule, or the Nemhauser-Trotter rule [31], are lop-kernels. More involved kernels, such as those based on protrusion replacement [9], are also lop-kernels. Hence, the most natural question is whether the "lop" assumption could be dropped from our general results, namely Theorem 3 and Theorem 13. For the vertex-minimization version (Theorem 13), we know that this is not possible: the problem of deleting at most k vertices from an n-vertex graph in order to obtain a tree admits a kernel with $\mathcal{O}(k^4)$ vertices [34], but no $\mathcal{O}(n^{1-\varepsilon})$-approximation for any $\varepsilon > 0$ unless $\mathsf{P} \neq \mathsf{NP}$ [49]. Therefore, if a polynomial lop-kernel for this problem existed, it would contradict Theorem 13, assuming that $\mathsf{P} \neq \mathsf{NP}$.

Thus, the algebraic reduction rule presented by Giannopoulou et al. [34], which is based on identifying a subset of linear equations of appropriate size that captures all solutions of size at most k, cannot be (even transformed to) a lop-rule. We still do not know of a similar example that is a vertex-maximization problem.

We showed that a direct application of Theorem 3 yields kernelization lower bounds for MMVC (Corollary 4) and MMFVS (Corollary 5), matching the sizes of the best known kernels for these problems. We believe that our result could be applied to other vertex-maximization problems, in particular to the "max-min" version of other vertex-minimization problems, as they seem to be quite hard to approximate. It would be interesting to find examples of vertex-minimization problems where Theorem 13 could be applied. Here, the natural candidates seem to be the "min-max" version of vertex-maximization problems, which seem to have been almost unexplored so far. We are currently working on the adaptation of our framework to edge-maximization (or minimization) problems, and even to problems with more general objective functions.

We presented (Section 6) subquadratic kernels on H-free graphs for some graphs H satisfying the (constructive) Erdős-Hajnal property, such as the bull, the complete graphs, or the paw. It would be interesting to obtain subquadratic kernels for other graphs H satisfying the Erdős-Hajnal property, such as C_4, the diamond, P_5, or C_5. Note that, from [38], C_4 and the diamond satisfy the constructive Erdős-Hajnal property with constant $\delta \geq 1/3$. Note also that the graphs constructed in the reduction of Theorem 17 are $\{C_5, \text{diamond}\}$-free, as they are bipartite, hence MMVC is NP-hard on this class, in contrast to the fact (see the full version [3]) that MMVC can be solved in linear time on $\{P_5, \text{diamond}\}$-free graphs. To the best of our knowledge, the complexity on P_5-free graphs is open, as well as on $K_{1,t}$ graphs for $t \geq 3$ (see the full version [3]). It is worth mentioning that P_5-free graphs have unbounded cliquewidth, because co-bipartite graphs, which are P_5-free, have unbounded cliquewidth.

As defined in Section 3, for a graph G we denoted by $\mathsf{mmvc}(G)$ the maximum size of a minimal vertex cover of G. Boria et al. [16] proved that if G is an n-vertex graph without isolated vertices, then $\mathsf{mmvc}(G) \geq \lfloor n^{1/2} \rfloor$. Note that this immediately yields a quadratic kernel for MMVC: if $k \leq \lfloor n^{1/2} \rfloor$ we answer "yes", otherwise $n \leq k^2$. By the same argument, if $\mathcal{C}$ is a graph class such that every n-vertex graph $G \in \mathcal{C}$ without isolated vertices satisfies $\mathsf{mmvc}(G) \geq n^{1/2+\varepsilon}$, for some $\varepsilon > 0$, then MMVC restricted to $\mathcal{C}$ admits a (subquadratic) kernel with at most $k^{\frac{2}{1+2\varepsilon}}$ vertices. It might be possible that this is the case for some of the H-free graph classes for which we provided subquadratic kernels in Section 6: we were not able to find any counterexample, that is, a family of n-vertex H-free graphs G for which $\mathsf{mmvc}(G) = \Theta(n^{1/2})$. In particular, the case of triangle-free graphs seems particularly interesting. Haviland [39] and Goddard and Lyle [36] established upper bounds on the size of a minimum independent dominating set (that is, the complement of a minimal vertex cover) of triangle-free graphs. It follows from their results [36, 39] that there exist n-vertex triangle-free graphs G with $\mathsf{mmvc}(G) = \Theta(n^{2/3} \cdot \log n)$, hence if such a constant $\varepsilon > 0$ as discussed above exists for triangle-free graphs, necessarily $\varepsilon \leq \frac{2}{3} - \frac{1}{2} = \frac{1}{6}$. Therefore, the smallest kernel that we may obtain in this way on triangle-free graphs would have $k^{\frac{2}{1+2\varepsilon}} \leq k^{3/2}$ vertices, which matches the size of the kernel that we obtained in the full version [3] for the particular case $t = 3$, disregarding lower-order terms and multiplicative constants. Finding such a constant $\varepsilon > 0$ on H-graphs for small graphs H, in particular on triangle-free graphs, looks like a challenging problem, having interesting connections with the Ramsey numbers [36, 39].

References

1 Faisal N. Abu-Khzam, Cristina Bazgan, Morgan Chopin, and Henning Fernau. Approximation algorithms inspired by kernelization methods. In *Proc. of the 25th International Symposium on Algorithms and Computation (ISAAC)*, volume 8889 of *LNCS*, pages 479–490, 2014. doi:10.1007/978-3-319-13075-0_38.

2 Noga Alon, János Pach, and József Solymosi. Ramsey-type theorems with forbidden subgraphs. *Combinatorica*, 21(2):155–170, 2001. doi:10.1007/s004930100016.

3 Júlio Araújo, Marin Bougeret, Victor A. Campos, and Ignasi Sau. A new framework for kernelization lower bounds: the case of Maximum Minimal Vertex Cover. Full version of the current article, 2021. arXiv:2102.02484.

4 Júlio Araújo, Marin Bougeret, Victor A. Campos, and Ignasi Sau. Parameterized complexity of computing maximum minimal blocking and hitting sets. Manuscript submitted for publication, 2021. arXiv:2102.03404.

5 Cristina Bazgan, Ljiljana Brankovic, Katrin Casel, Henning Fernau, Klaus Jansen, Kim-Manuel Klein, Michael Lampis, Mathieu Liedloff, Jérôme Monnot, and Vangelis Th. Paschos. The many facets of upper domination. *Theoretical Computer Science*, 717:2–25, 2018. doi:10.1016/j.tcs.2017.05.042.

6 Daniel Binkele-Raible, Henning Fernau, Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Yngve Villanger. Kernel(s) for problems with no kernel: On out-trees with many leaves. *ACM Transactions on Algorithms*, 8(4):38:1–38:19, 2012. doi:10.1145/2344422.2344428.

7 Arindam Biswas, Venkatesh Raman, and Saket Saurabh. Approximation in (poly-) logarithmic space. In *Proc. of the 45th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 170 of *LIPIcs*, pages 16:1–16:15, 2020. doi:10.4230/LIPIcs.MFCS.2020.16.

8 Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *Journal of Computer and System Sciences*, 75(8):423–434, 2009. doi:10.1016/j.jcss.2009.04.001.

9 Hans L. Bodlaender, Fedor V. Fomin, Daniel Lokshtanov, Eelko Penninkx, Saket Saurabh, and Dimitrios M. Thilikos. (Meta) Kernelization. *Journal of the ACM*, 63(5):44:1–44:69, 2016.

10 Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. Kernelization lower bounds by cross-composition. *SIAM Journal on Discrete Mathematics*, 28(1):277–305, 2014. doi:10.1137/120880240.

11 Hans L. Bodlaender, Stéphan Thomassé, and Anders Yeo. Kernel bounds for disjoint cycles and disjoint paths. *Theoretical Computer Science*, 412(35):4570–4578, 2011. doi:10.1016/j.tcs.2011.04.039.

12 Rodica Boliac and Vadim V. Lozin. Independent domination in finitely defined classes of graphs. *Theoretical Computer Science*, 301(1-3):271–284, 2003. doi:10.1016/S0304-3975(02)00586-8.

13 Édouard Bonnet, Michael Lampis, and Vangelis Th. Paschos. Time-approximation trade-offs for inapproximable problems. *Journal of Computer and System Sciences*, 92:171–180, 2018. doi:10.1016/j.jcss.2017.09.009.

14 Édouard Bonnet and Vangelis Th. Paschos. Sparsification and subexponential approximation. *Acta Informatica*, 55(1):1–15, 2018. doi:10.1007/s00236-016-0281-2.

15 Édouard Bonnet, Stéphan Thomassé, Xuan Thang Tran, and Rémi Watrigant. An algorithmic weakening of the Erdős-Hajnal conjecture. In *Proc. of the 28th Annual European Symposium on Algorithms (ESA)*, volume 173 of *LIPIcs*, pages 23:1–23:18, 2020. doi:10.4230/LIPIcs.ESA.2020.23.

16 Nicolas Boria, Federico Della Croce, and Vangelis Th. Paschos. On the max min vertex cover problem. *Discrete Applied Mathematics*, 196:62–71, 2015. doi:10.1016/j.dam.2014.06.001.

17 Katrin Casel, Henning Fernau, Mehdi Khosravian Ghadikolaei, Jérôme Monnot, and Florian Sikora. Extension of Vertex Cover and Independent Set in Some Classes of Graphs. In *Proc. of the 11th International Conference on Algorithms and Complexity (CIAC)*, volume 11485 of *LNCS*, pages 124–136, 2019. doi:10.1007/978-3-030-17402-6_11.

18 Jianer Chen, Henning Fernau, Iyad A. Kanj, and Ge Xia. Parametric duality and kernelization: Lower bounds and upper bounds on kernel size. *SIAM Journal on Computing*, 37(4):1077–1106, 2007. doi:10.1137/050646354.

19 Maria Chudnovsky. The Erdős-Hajnal Conjecture - A Survey. *Journal of Graph Theory*, 75(2):178–190, 2014. doi:10.1002/jgt.21730.

20 Maria Chudnovsky and Shmuel Safra. The Erdős-Hajnal conjecture for bull-free graphs. *Journal of Combinatorial Theory, Series B*, 98(6):1301–1310, 2008. doi:10.1016/j.jctb.2008.02.005.

21 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

22 Peter Damaschke. Parameterized algorithms for double hypergraph dualization with rank limitation and maximum minimal vertex cover. *Discrete Optimization*, 8(1):18–24, 2011. doi:10.1016/j.disopt.2010.02.006.

23 Holger Dell and Dániel Marx. Kernelization of packing problems. In *Proc. of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 68–81, 2012. doi:10.1137/1.9781611973099.6.

24 Holger Dell and Dieter van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. *Journal of the ACM*, 61(4):23:1–23:27, 2014. doi:10.1145/2629620.

25 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012. URL: https://dblp.org/rec/books/daglib/0030488.bib.

26 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

27 Louis Dublois, Tesshu Hanaka, Mehdi Khosravian Ghadikolaei, Michael Lampis, and Nikolaos Melissinos. (In)approximability of Maximum Minimal FVS. In *Proc. of the 31st International Symposium on Algorithms and Computation (ISAAC)*, volume 181 of *LIPIcs*, pages 3:1–3:14, 2020. The cubic kernel appears in the full version, available at https://arxiv.org/abs/2009.09971. doi:10.4230/LIPIcs.ISAAC.2020.3.

28 Louis Dublois, Michael Lampis, and Vangelis Th. Paschos. Upper Dominating Set: Tight Algorithms for Pathwidth and Sub-Exponential Approximation. *CoRR*, abs/2101.07550, 2021. arXiv:2101.07550.

29 Paul Erdős and András Hajnal. Ramsey-type theorems. *Discrete Applied Mathematics*, 25(1-2):37–52, 1989. doi:10.1016/0166-218X(89)90045-0.

30 Henning Fernau. *Parameterized algorithms: a graph-theoretic approach*. Habilitationsschrift, Universität Tübingen, 2005. URL: http://www.informatik.uni-trier.de/~fernau/papers/habil.pdf.

31 Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization. Theory of Parameterized Preprocessing*. Cambridge University Press, 2019. doi:10.1017/9781107415157.

32 Lance Fortnow and Rahul Santhanam. Infeasibility of instance compression and succinct PCPs for NP. *Journal of Computer and System Sciences*, 77(1):91–106, 2011. doi:10.1016/j.jcss.2010.06.007.

33 Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979. doi:https://dl.acm.org/doi/book/10.5555/574848.

34 Archontia C. Giannopoulou, Daniel Lokshtanov, Saket Saurabh, and Ondrej Suchý. Tree Deletion Set has a polynomial kernel but no $\mathrm{OPT}^{O(1)}$ approximation. *SIAM Journal on Discrete Mathematics*, 30(3):1371–1384, 2016. doi:10.1137/15M1038876.

35 Wayne Goddard and Michael A. Henning. Independent domination in graphs: A survey and recent results. *Discrete Mathematics*, 313(7):839–854, 2013. doi:10.1016/j.disc.2012.11.031.

36 Wayne Goddard and Jeremy Lyle. Independent dominating sets in triangle-free graphs. *Journal of Combinatorial Optimization*, 23(1):9–20, 2012. doi:10.1007/s10878-010-9336-4.

37 Jiong Guo, Iyad A. Kanj, and Stefan Kratsch. Safe approximation and its relation to kernelization. In *Proc. of the 6th International Symposium on Parameterized and Exact Computation (IPEC)*, volume 7112 of *LNCS*, pages 169–180, 2011. doi:10.1007/978-3-642-28050-4_14.

38 András Gyárfás. Reflections on a Problem of Erdős and Hajnal. In Ronald L. Graham, Jaroslav Nesetril, and Steve Butler, editors, *The Mathematics of Paul Erdős II*, pages 135–141. Springer, 2013. doi:10.1007/978-1-4614-7254-4_11.

39 Julie Haviland. Independent domination in triangle-free graphs. *Discrete Mathematics*, 308(16):3545–3550, 2008. doi:10.1016/j.disc.2007.07.010.

40 Danny Hermelin and Xi Wu. Weak compositions and their applications to polynomial lower bounds for kernelization. In *Proc. of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 104–113, 2012. doi:10.1137/1.9781611973099.9.

41 Dorit S. Hochbaum and David B. Shmoys. Using dual approximation algorithms for scheduling problems theoretical and practical results. *Journal of the ACM*, 34(1):144–162, 1987. doi:10.1145/7531.7535.

42 Johann L. Hurink and Tim Nieberg. Approximating minimum independent dominating sets in wireless networks. *Information Processing Letters*, 109(2):155–160, 2008. doi:10.1016/j.ipl.2008.09.021.

43 Stefan Kratsch. Polynomial kernelizations for MIN $F^+\Pi_1$ and MAX NP. *Algorithmica*, 63(1-2):532–550, 2012. doi:10.1007/s00453-011-9559-5.

44 Stefan Kratsch, Marcin Pilipczuk, Ashutosh Rai, and Venkatesh Raman. Kernel lower bounds using co-nondeterminism: Finding induced hereditary subgraphs. *ACM Transactions on Computation Theory*, 7(1):4:1–4:18, 2014. doi:10.1145/2691321.

45 Daniel Lokshtanov, Fahad Panolan, M. S. Ramanujan, and Saket Saurabh. Lossy kernelization. In *Proc. of the 49th Annual ACM Symposium on Theory of Computing (STOC)*, pages 224–237, 2017. doi:10.1145/3055399.3055456.

46 Oystein Ore. *Theory of Graphs*. American Mathematical Society Colloquium Publications, volume 38, 1962. URL: https://bookstore.ams.org/coll-38.

47 Neil Robertson, Daniel P. Sanders, Paul D. Seymour, and Robin Thomas. The four-colour theorem. *Journal of Combinatorial Theory, Series B*, 70(1):2–44, 1997. doi:10.1006/jctb.1997.1750.

48 David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, 2011. URL: http://www.cambridge.org/de/knowledge/isbn/item5759340/?site_locale=de_DE.

49 Mihalis Yannakakis. The effect of a connectivity requirement on the complexity of maximum subgraph problems. *Journal of the ACM*, 26(4):618–630, 1979. doi:10.1145/322154.322157.

50 Meirav Zehavi. Maximum Minimal Vertex Cover Parameterized by Vertex Cover. *SIAM Journal on Discrete Mathematics*, 31(4):2440–2456, 2017. doi:10.1137/16M109017X.