Feedback Vertex Set for pseudo-disk graphs in subexponential FPT time

Gaétan Berthe[1] [0000−0003−0017−6922], Marin Bougeret[1] [0000−0002−9910−4656], Daniel Gonçalves[1] [0000−0003−3228−9622], and Jean-Florent Raymond[2] [0000−0003−4646−7602]

[1] LIRMM, Université de Montpellier, CNRS, Montpellier, France.
[2] CNRS, LIP, Lyon, France.

Abstract. In this paper we investigate the existence of parameterized algorithms running in subexponential time for two fundamental cycle-hitting problems: Feedback Vertex Set and Triangle Hitting. We focus on the class of pseudo-disk graphs, which forms a common generalization of several graph classes where such results exist, like disk graphs and square graphs. In these graphs we show that given a geometric representation FVS can be solved in time $2^{\mathcal{O}(k^{9/10} \log k)} n^{\mathcal{O}(1)}$ and TH in time $2^{\mathcal{O}(k^{3/4} \log k)} n^{\mathcal{O}(1)}$.

Keywords: geometric intersection graphs, subexponential FPT algorithms, cycle-hitting problems, pseudo-disk graphs

1 Introduction

Context. The purpose of Parameterized Complexity is to provide an accurate view of the algorithmic complexity of a (typically NP-hard) decision problem and to understand the different contributions to the running time of the parameters of the instance. In this framework, a standard objective is to find FPT algorithms whose time complexities have the form $f(k) \cdot n^{\mathcal{O}(1)}$ where n and k respectively denote the size and some parameter of the instance, and f is some computable function. Hence, the potential super-polynomial part of the running time is confined to the $f(k)$ term.

In this paper we mainly focus on Feedback Vertex Set (FVS for short) which is the problem of deciding, given a graph G and an integer k, whether G has a set of k vertices whose deletion yields a forest. Some NP-hard problems like FVS cannot be solved in time $2^{o(k)} n^{\mathcal{O}(1)}$ in general graphs [3] (assuming the Exponential Time Hypothesis) but nevertheless admit algorithms with such running times (called *subexponential FPT algorithms*) when the inputs are restricted to certain graph classes. This was initially proved for particular problems in planar graphs and related classes (like map graphs) and later unified by Demaine el al. [4] in a general framework called *Bidimensionality Theory*, which in essence states that every *bidimensional*[3] problem (like FVS) can be solved in subexponential FPT time on any graph class excluding a minor.

The next step has then been to move away from minor-closed graph classes and investigate in which other classes the basic NP-hard graph problems admit subexponential FPT algorithms. Natural candidates in this direction are intersection graphs of objects in the plane. Indeed, while such graphs are not planar, the underlying planarity may allow to lift techniques and ideas from the bidimensionality theory. This is not straightforward in general and required new ideas as explained for example in [12] which discusses the extension of bidimensionality to (unit) disk graphs.

These recent developments led to subexponential FPT algorithms in disk graphs [12] running in time $2^{\mathcal{O}(k^{13/14}\log k)}n^{\mathcal{O}(1)}$ for FVS, and in $2^{\mathcal{O}(k^{9/10}\log k)}n^{\mathcal{O}(1)}$ for TH (the Triangle Hitting problem where given a graph G and integer k one has to decide if there is a set S of k vertices whose deletion yields to a triangle-free graph). Recently [1], these running times have been improved to $2^{\mathcal{O}(k^{7/8}\log k)}n^{\mathcal{O}(1)}$ for FVS and $2^{\mathcal{O}(k^{2/3}\log k)}n^{\mathcal{O}(1)}$ for TH when a disk representation is given, and a $2^{\mathcal{O}(k^{9/10}\log k)}n^{\mathcal{O}(1)}$ for FVS and $2^{\mathcal{O}(k^{4/5}\log k)}n^{\mathcal{O}(1)}$ for TH otherwise. On the other hand, we proved in a companion paper [2] that under the Exponential Time Hypothesis, neither TH nor FVS do admit algorithms running in time $2^{o(\sqrt{n})}$ in $K_{2,2}$-free contact-2-DIR graphs of maximum degree 6, a very restricted subclass of pseudo-disk graphs.

Contribution and techniques. In this paper we consider pseudo-disk graphs, a classical generalization of disk graphs where informally each vertex is now a pseudo-disk (a subset of the plane that is homeomorphic to a disk), and such that for any two intersecting pseudo-disks, their boundaries intersect on at most two points. We prove that, given a pseudo-disk representation of a graph, one can solve FVS in time $2^{\mathcal{O}(k^{9/10}\log k)}n^{\mathcal{O}(1)}$ (Theorem 2), which is our main result, and TH in time $2^{\mathcal{O}(k^{3/4}\log k)}n^{\mathcal{O}(1)}$.

For FVS, we apply the following strategy. Given an input (G, k), the objective is to reduce the treewidth of G to $o(k)$ in order to solve FVS in $2^{\mathcal{O}(\mathsf{tw}(G))}n^{\mathcal{O}(1)}$ using a standard dynamic programming algorithm [3]. To do so, one has to delete from G any structure that could make $\mathsf{tw}(G)$ large (close to k). A classical approach is to start from a $\mathcal{O}(1)$-approximation M_0 (with $|M_0| = \mathcal{O}(k)$) of a minimum FVS. Observe that as FVS is bidimensional, the existence of M_0 implies that $\boxplus(G)$ (the size of the largest square grid contained as a minor in G) is $\mathcal{O}(\sqrt{k})$. However, unlike in planar graphs where $\mathsf{tw}(G) = \mathcal{O}(\boxplus(G))$ (a property known under the name of *subquadratic grid minor property*), having $\boxplus(G) = \mathcal{O}(\sqrt{k})$ is not enough to ensure that $\mathsf{tw}(G) = o(k)$, as for example a clique K_k has $\boxplus(K_k) \leq \sqrt{k}$ but $\mathsf{tw}(K_k) = k - 1$. Thus, as first step, we use a folklore preprocessing routine that produces $2^{\mathcal{O}\left(\frac{k}{p}\log p\right)}n^{\mathcal{O}(1)}$ instances (G', k') where each graph has clique number at most $p = \mathcal{O}(k^{\epsilon})$.

[3] Informally: yes-instances are minor-closed and a solution on the (r, r)-grid has size $\Omega(r^2)$.

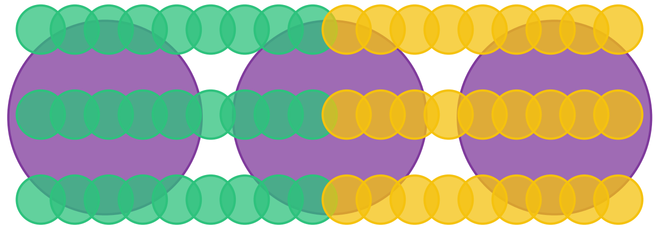

Fig. 1. In this graph G' (from [8]), $\omega(G')$ is constant, $\mathsf{tw}(G') \geq k'$ (where $k' = 3$ here) as it contains $K_{k',k'}$ as a minor, and $\boxplus(G) = \mathcal{O}(\sqrt{k'})$ as the set M_0 of the k' big disks is a FVS.

The problem now is that (even in disk-graphs) this additional property on the clique number is still not enough to force small treewidth, as shown for instance by the graph G' of Figure 1. Thus, we apply a second preliminary branching step (described in the context of disk graphs in [12]). This produces again $2^{\mathcal{O}\left(\frac{k}{p}\log p\right)} n^{\mathcal{O}(1)}$ instances (G'', M, k''), where M is a feedback vertex set of G'' with $|M| = \mathcal{O}(pk') = \mathcal{O}(pk)$, with the additional property that for any $v \in M$, $N(v) \setminus M$ is an independent set. This property may look like a technical assumption, but observe that it is necessary to ensure that the treewidth is $o(k)$. For example with G' and M_0 like in Figure 1, we have $\mathsf{tw}(G') \geq k'$, but this case is no longer possible after the second branching step as the property that for any $v \in M_0$, $N(v) \setminus M_0$ is an independent set does *not* hold.

Once these two preliminaries branching steps are done, our main technical challenge is to kernelize each instance (G'', M, k'') to obtain a smaller equivalent instance $\left(\tilde{G}, \tilde{k}\right)$ such that $\left|V\left(\tilde{G}\right)\right| = \mathcal{O}\left(p^6|M|\right) = \mathcal{O}\left(p^7 k\right)$ (see Lemma 2). This is the crux of the paper. The techniques used to achieve this kernel are sketched in Section 4. Once the number of vertices of $\tilde{G}$ is small, we can show (Lemma 1) that $\mathsf{tw}(\tilde{G}) = \mathcal{O}\left(\sqrt{p\left|V\left(\tilde{G}\right)\right|}\right) = \mathcal{O}\left(\sqrt{k^{1+8\epsilon}}\right)$ and solve $\tilde{G}$ by classic dynamic programming, leading to our subexponential algorithm for FVS in pseudo-disk graphs.[4]

For TH, we revisit the strategy of [1] (originally designed for disk graphs) that produces a subexponential number of instances of order quasi-linear in the parameter. To improve the running time we show (Lemma 1) that the number of edges in a n-vertex pseudo-disk graph of ply[5] p is $\mathcal{O}(pn)$, which implies that the produced instances have sublinear treewidth $\mathcal{O}(\sqrt{pn})$. This improves the treewidth bound $f(p)\sqrt{n}$ (for some function f) given by Dvorák et al. in [6]. The problem is then solved by dynamic programming on approximate tree-decompositions.

[4] We have been told in a private communication that it might be possible to extend the arguments of [12] for FVS in pseudo-disk graphs without a geometrical representation, and that the time complexity would be worse than the one we obtain.

[5] The ply is the maximum number of pseudo-disks sharing a common point.

Organization of the paper. In Section 2 we give the necessary definitions and properties and present the aforementioned preprocessing step. Section 3 reduces our main result to a technical kernelization lemma. Section 4 is devoted to sketching the first part of the proof of this lemma. We conclude with directions for future research in Section 5. Due to space constraints, the proofs of the statements marked with the ✂ symbol can be found in the full version of the paper.

2 Preliminaries

2.1 Basics

In this paper logarithms are binary and all graphs are simple, loopless and undirected. Unless otherwise specified we use standard graph theory terminology, as in [5] for instance. Given a graph G, we denote by $\omega(G)$ the maximum order of a clique in G. We denote by $d_G(v)$ the degree of $v \in V(G)$, or simply $d(v)$ when G is clear from the context. We use the notation $\Delta(G)$ for the maximum degree of the vertices of G. We denote by $\mathsf{tw}(G)$ the treewidth of G.

2.2 Graph classes

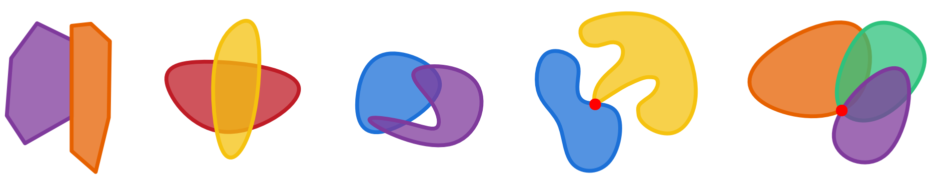

Fig. 2. Three forbidden intersections between pseudo-disks, and two avoidable examples.

In this article, we are mainly concerned with geometric graphs described by the intersection or contact of objects in the Euclidean plane. The most general class we consider are *string graphs*, which are intersection graphs of strings (a.k.a. Jordan arcs).

Pseudo-disk graphs. The focus on this paper is on *pseudo-disk graphs*. A *pseudo-disk* $\mathcal{D}$ is a subset of the plane that is homeomorphic to a disk. We denote its boundary by $\partial\mathcal{D}$ and call *internal points* any point of $\mathcal{D}$ that does not belong to $\partial\mathcal{D}$. A set of pseudo-disks $\mathcal{S}$ forms a *system of pseudo-disks* if, for any two intersecting elements $\mathcal{D}_1, \mathcal{D}_2 \in \mathcal{S}$,

their borders, $\partial\mathcal{D}_1$ and $\partial\mathcal{D}_2$, intersect on at most two points. Under minor perturbation, any system of pseudo-disk, can be such that for any two intersecting elements $\mathcal{D}_1, \mathcal{D}_2 \in \mathcal{S}$, either their borders do not intersect but in that case one is contained in the other, or their borders intersect on exactly two points while $\mathcal{D}_1 \cap \mathcal{D}_2$ contains internal points. Similarly, we can require that no point belongs to more than two boundaries (see Figure 2). Given a system of pseudo-disks $\mathcal{S}$, we denote by $G_{\mathcal{S}}$ the corresponding intersection graph, and this defines the class of pseudo-disk graphs.

Note that pseudo-disk graphs are in particular string graphs and they form a common generalization of various classes of "fat" intersection graphs such as disk graphs, intersection graphs of axis-parallel squares, and more generally any intersection graph obtained from homothetic copies of a given convex set, but they also generalize the contact graphs of segments (see full version).

Given $\mathcal{S}$ a system of pseudo-disks and z a point in the plane, the *ply* of z (wrt. $\mathcal{S}$) is the number of pseudo-disks of $\mathcal{S}$ containing z. The *ply* of a maximal connected region $\mathcal{R}$ of $\mathbb{R}^2 \setminus \bigcup_{\mathcal{D}\in\mathcal{S}} \partial\mathcal{D}$, is the ply of any of its points. The *ply* of $\mathcal{S}$ is the maximum ply of a point of the plane wrt. $\mathcal{S}$. A pseudo-disk graph G has *ply* p if it is the intersection graph of a system of pseudo-disks of ply p.

Representation of pseudo-disk graphs. A system of pseudo-disks $\mathcal{S}$ is represented by the directed plane multi-graph $\overrightarrow{P}_{\mathcal{S}}$ defined as follows (see also Figure 3). For any two $\mathcal{D}_1, \mathcal{D}_2 \in \mathcal{S}$, every point in $\partial\mathcal{D}_1 \cap \partial\mathcal{D}_2$ is a vertex of $\overrightarrow{P}_{\mathcal{S}}$. For any $\mathcal{D} \in \mathcal{S}$, the Jordan arcs in $\partial\mathcal{D}$ joining any two such points form the arcs of $\overrightarrow{P}_{\mathcal{S}}$. Those arcs are oriented in such a way that $\partial\mathcal{D}$ corresponds to a clockwise cycle around $\mathcal{D}$. It may remain disks $\mathcal{D} \in \mathcal{S}$ with uncrossed boundaries (i.e. $\partial\mathcal{D} \cap \partial\mathcal{D}' = \emptyset$ for any $\partial\mathcal{D}' \in \mathcal{S}$). For such a disk, we pick an arbitrary point in $\partial\mathcal{D}$ as a vertex for $\overrightarrow{P}_{\mathcal{S}}$, and the rest of $\partial\mathcal{D}$ corresponds to a clockwise loop on this vertex. Note that $\overrightarrow{P}_{\mathcal{S}}$ has at most $2|E(G_{\mathcal{S}})| + |V(G_{\mathcal{S}})|$ vertices, and at most $4|E(G_{\mathcal{S}})| + |V(G_{\mathcal{S}})|$ edges. This is a convenient feature of pseudo-disks system to admit a polynomial (in terms of $G_{\mathcal{S}}$) space data structure representing them. This graph $\overrightarrow{P}_{\mathcal{S}}$ is called the *pseudo-disk representation* of $\mathcal{S}$ and $G_{\mathcal{S}}$. We denote $\overrightarrow{P}^*_{\mathcal{S}}$ the dual graph of $\overrightarrow{P}_{\mathcal{S}}$. Observe that any arc is oriented from a region (of $\mathcal{S}$) with lower ply towards one with higher ply.

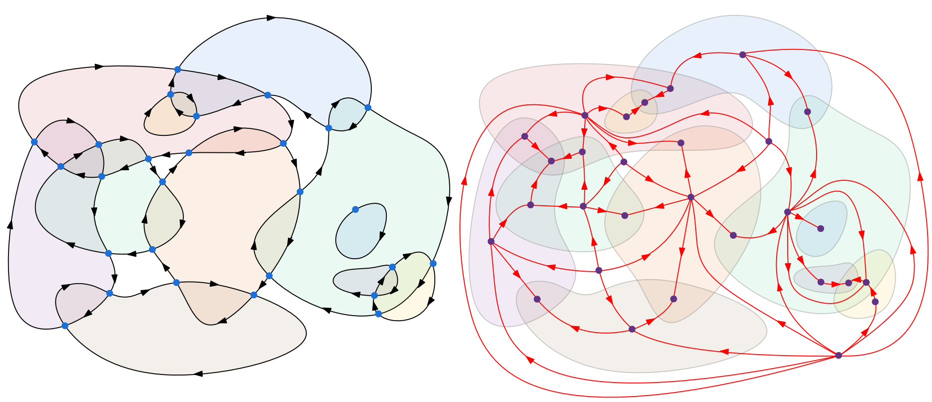

Fig. 3. A pseudo-disk representation $\overrightarrow{P}_{\mathcal{S}}$ and its dual $\overrightarrow{P}^*_{\mathcal{S}}$.

2.3 Branching steps and properties of pseudo-disk graphs

Dvořák et al. proved in [6] that n-vertex pseudo-disk graphs with ply p have treewidth at most $f(p)\sqrt{n}$ for some function f. In Lemma 1 below we improve this bound to $\mathcal{O}(\sqrt{pn})$. Our proof requires the following consequence of the results on balanced separators of string graphs of Lee [11] and the links between separators and treewidth of Dvořák and Norin [7].

Theorem 1 ([11] and [7]). *Any m-edge string graph has treewidth $\mathcal{O}(\sqrt{m})$.*

Lemma 1. *If a graph G on n vertices admits a pseudo-disk representation with ply p, then G has at most $3epn$ edges. Furthermore, $\mathsf{tw}(G) = \mathcal{O}(\sqrt{pn})$.*

Proof. To bound the number of edges of G, we follow the same approach as [9] for contact of strings. Consider a pseudo-disk graph G with n vertices and m edges having a representation with ply p. Then, pick a subset $V' \subseteq V(G)$ of pseudo-disks by picking each one, randomly and independently, with probability $1/p$. The expected size of V' is n/p. For each $uv \in E(G)$, the probability that at an arbitrarily chosen point $p_{uv} \in \mathcal{D}_u \cap \mathcal{D}_v$, only $\mathcal{D}_u$ and $\mathcal{D}_v$ remain is $(1-1/p)^{q-2}/p^2 \geq (1-1/p)^{p-2}/p^2$, where q is the ply of p_{uv}. By linearity of expectation, the expected number of edges uv for which only u and v remain at p_{uv} is at least $m(1-1/p)^{p-2}/p^2$. The following claim implies that $m(1-1/p)^{p-2}/p^2 \leq 3n/p$.

Claim 1. If a graph H has an edge subset $E' \subseteq E(H)$ such that H admits a pseudo-disk representation where for any edge $uv \in E'$ there is a point in $\mathcal{D}_u \cap \mathcal{D}_v$ with ply two, then $|E'| \leq 3|V(H)|$.

Proof. We show this by proving that the graph $(V(H), E')$ is planar, and this is obtained by constructing a pseudo-disk representation with ply at most two [10]. We start with the representation of H and we show how to modify it in order to delete at least one edge of $E(H) \setminus E'$. Iterating this process yields the desired representation.

If there is a pseudo-disk $\mathcal{D}_v$, whose points all have ply at least three, one can simply move $\mathcal{D}_v$ far away from the rest of the representation. The obtained graph has at least two edges less, and we still have a point with ply two in $\mathcal{D}_u \cap \mathcal{D}_v$ for any $uv \in E'$.

Similarly if there is an edge ab of $E(H) \setminus E'$ with $\mathcal{D}_a \subset \mathcal{D}_b$ and a has degree 1, we can move its disk away from the rest of the representation.

Otherwise, let ab be an edge of $E(H) \setminus E'$ minimizing the number of regions in $\mathcal{D}_a \cap \mathcal{D}_b$. Note that by the first rule there is no $\mathcal{D}_c \subseteq \mathcal{D}_a \cap \mathcal{D}_b$. For such an edge ab, and for every $c \in V(H) \setminus \{a, b\}$, the border $\partial\mathcal{D}_c$ intersects $\partial\mathcal{D}_a \cap \mathcal{D}_b$ as many times as it intersect $\partial\mathcal{D}_b \cap \mathcal{D}_a$. Hence, replacing $\mathcal{D}_a$ and $\mathcal{D}_b$ by $\mathcal{D}_a \setminus \mathcal{D}_b$ and $\mathcal{D}_b \setminus \mathcal{D}_a$ respectively, one obtains a representation where the new borders $\partial\mathcal{D}_a$ and $\partial\mathcal{D}_b$ intersect the same borders (exactly twice) as the old one except for the edge ab. Finally, since there was no pseudo-disk contained in $\mathcal{D}_a \cap \mathcal{D}_b$, ab is the only deleted edge. ⌟

Since $1/e \leq (1-1/p)^{p-2}$ for $p \geq 3$, we get $m \leq 3epn$ as claimed. The second part of the statement follows by Theorem 1. $\square$

As first step of our algorithms we make use of two preprocessing routines: the first one is a folklore branching that allows to reduce cliques larger than a chosen size p (where typically $p = k^\epsilon$) and the second, given a (possibly non-optimal) solution M to the problem, allows to consider subcases where for any $v \in M$, $N(v) \setminus M$ is an independent set. These steps are described in [12] for disk graphs. We combine them in the following routine.

Corollary 1 (✂). *Let Π be FVS or TH. There is a $2^{\mathcal{O}\left(\frac{k}{p}\log k\right)} n^{\mathcal{O}(1)}$-time algorithm that, given a pseudo-disk graph G with a representation, a parameter k, and an integer $p \in [6,k]$, returns a collection $\mathcal{C}$ of size $2^{\mathcal{O}\left(\frac{k}{p}\log k\right)}$ of quadruples $(G', \overrightarrow{P'}, M, k')$ such that $\overrightarrow{P'}$ is a pseudo-disk representation of G' and:*

1. *(G', k') is an instance of Π where G' is an induced subgraph of G, $\omega(G') \leq p$, and $k' \leq k$.*
2. *$|M| = \mathcal{O}(pk)$, $G' - M$ is triangle-free (when Π is TH) or is a forest (when Π is FVS), and for any $v \in M$, $N(v) \setminus M$ is an independent set.*
3. *(G, k) is a yes-instance of Π if and only if there exists $(G', \overrightarrow{P'}, M, k') \in \mathcal{C}$ such that (G', k') is a yes-instance.*

3 Hitting cycles in pseudo-disk graphs

The main result of this paper is the following.

Theorem 2. *There is an algorithm that, given an n-vertex pseudo-disk graph with a representation and a parameter k, solves FVS in time $2^{\mathcal{O}(k^{\frac{9}{10}}\log k)} n^{O(1)}$.*

To achieve this, we proceed in three steps. We first use the preprocessing of Corollary 1. Then, we kernelize each instance provided by the branching process in Lemma 2. We show that these kernelized instances have small treewidth, and thus we can conclude with a dynamic algorithm to solve each of them. The main technical ingredient of this proof is the following lemma, whose proof is deferred to the full version due to space constraints, and for which we provide a sketch in Section 4.

Lemma 2 (✂). *Given a quadruple $(G, \overrightarrow{P}_{\mathcal{S}}, M, k)$ as given by Corollary 1 (for $\Pi = FVS$), there is a polynomial time algorithm that returns an equivalent instance (G', k') where $k' \leq k$ and G' is a pseudo-disk graph with ply at most p and $\mathcal{O}(p^7 k)$ vertices.*

As we will apply Corollary 1 for $p = \mathcal{O}(k^\epsilon)$, the above will directly imply that $|V(G')|$ is almost linear, and, using Lemma 1, that G' has sublinear treewidth. Thus, the whole technical difficulty is to prove Lemma 2. Now that the lemma is stated, we can proceed with the proof of the theorem.

Proof (of Theorem 2). We apply Corollary 1 on an instance (G, k) provided with a representation $\overrightarrow{P}_{\mathcal{S}}$ and on a value $p \in [6, k]$ to be set later. As a result we obtain a collection $\mathcal{C}$ of $2^{\mathcal{O}\left(\frac{k}{p}\log k\right)}$ instances that are each provided with a ply p pseudo-disk representation, and with a feedback vertex set M of size $\mathcal{O}(pk)$, such that for any $v \in M$, $N(v) \setminus M$ is an independent set. Furthermore, solving these instances of FVS is enough to get a solution to our initial instance (G, k). This first stage is done in time $2^{\mathcal{O}\left(\frac{k}{p}\log k\right)} n^{O(1)}$.

Then for each of the $2^{\mathcal{O}\left(\frac{k}{p}\log k\right)}$ obtained instances, we apply the kernelization described in Lemma 2, and we get a pseudo-disk graph G' with ply at most p and with only $\mathcal{O}(p^7 k)$ vertices. By Lemma 1, this kernel has treewidth at most $\mathcal{O}(\sqrt{p^8 k})$, and can thus be solved in time $2^{\mathcal{O}(\mathsf{tw}(G'))} n^{\mathcal{O}(1)} = 2^{\mathcal{O}\left(p^4 k^{\frac{1}{2}}\right)} n^{\mathcal{O}(1)}$ with classical algorithms [3]. The overall time complexity of this algorithm is

$$2^{\mathcal{O}\left(\frac{k}{p}\log k\right)} n^{O(1)} \cdot \, 2^{\mathcal{O}\left(\frac{k}{p}\log k\right)} \cdot n^{\mathcal{O}(1)} \cdot 2^{\mathcal{O}\left(p^4 k^{\frac{1}{2}}\right)} n^{O(1)}.$$

Thus setting $p = k^{\frac{1}{10}}$ the time complexity is indeed $2^{\mathcal{O}\left(k^{9/10}\log k\right)} n^{O(1)}$. □

4 Overview of the first part of proof of Lemma 2

Recall that in Lemma 2 we are given a quadruple $(G, \overrightarrow{P}_{\mathcal{S}}, M, k)$ as computed by Corollary 1. The equivalent instance computed in Lemma 2 will be obtained using reduction rules. Before applying these rules, we define the set M' by adding to M every pseudo-disk $\mathcal{D}$ that contains a point of $\partial\mathcal{D}_u \cap \partial\mathcal{D}_v$, for $u, v \in M$ (see Figure 4, left). We denote $\mathcal{R}_{M'}$ the region $\mathcal{R}_{M'} = (\cup_{u \in M'} \mathcal{D}_u)$. Note that this region (and its complement) may not be connected. Let us now partition the vertices of $V(G) \setminus M'$ into three types.

Definition 1. *A vertex $v \in V(G) \setminus M'$ is an* inner-M'-vertex *if $\mathcal{D}_v \subseteq \mathcal{R}_{M'}$, it is a* border-$M'$-vertex *if $\mathcal{D}_v$ intersects $\mathcal{R}_{M'}$ without being included, and it is an* outer-M'-vertex *if $\mathcal{D}_v$ does not intersect $\mathcal{R}_{M'}$. We denote $I_{M'}$, $B_{M'}$, and $O_{M'}$ the vertex sets with these three types of vertices (see Figure 4 (right)).*

Let us state a few properties about these vertices.

Claim 2 (✂).

(a) $|M'| = \mathcal{O}(p|M|)) = \mathcal{O}(p^2 k)$.
(b) For any edge uv of $G - M'$, the intersection $\mathcal{D}_u \cap \mathcal{D}_v$ does not intersect $\mathcal{R}_{M'}$.
(c) For every $u \in V(G) \setminus M'$, the pseudo-disk $\mathcal{D}_u$ does not contain any point of $\partial\mathcal{D}_v \cap \partial\mathcal{D}_w$, for any two vertices $v, w \in M'$.

Note that Property (b) implies that every inner-M'-vertex u is isolated in $G - M'$. The kernelization is divided into two parts. The first part (see Subsection 4.1) deals with these inner-M'-vertices, while the second part (not included here due to lack of space) deals with border- and outer-M'-vertices.

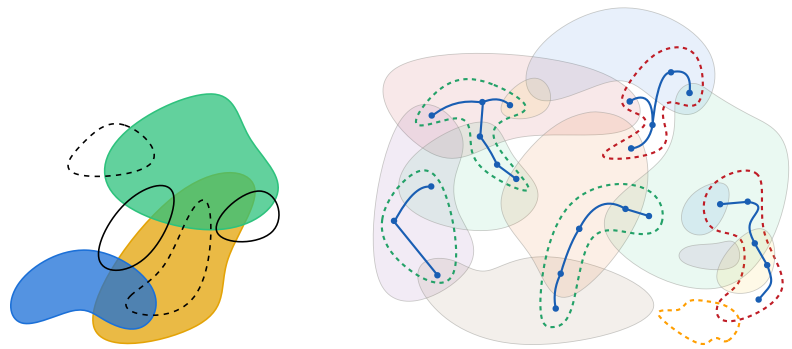

Fig. 4. (left) From M to M'. The pseudo-disks of M are filled. Among the others, those added to M' have solid border. (right) Here, the filled pseudo-disks belong to M'. In dashed green, the inner pseudo-disks of $I_{M'}$. In dashed red, the border pseudo-disks of $B_{M'}$. In dashed orange, the only pseudo-disk $v \in O_{M'}$. For any pseudo-disk u not in M', we depicted its hosted graph H_u, which is defined in Definition 2.

4.1 Sketch of proof for inner-M'-vertices

The purpose of this section is to reduce the number of inner-M'-vertices. Let $\mathcal{S}'$ be the restriction of $\mathcal{S}$ to vertices of M'. Let us denote $\overrightarrow{P}_{\mathcal{S}'}$ the representation of $G[M']$. Since the pseudo-disk system has ply at most p, the graph $G[M']$ has $\mathcal{O}(p|M'|) = \mathcal{O}(p^3k)$ edges by Lemma 1. Each of such edge induces at most two vertices in $\overrightarrow{P}_{\mathcal{S}'}$. Hence, $\overrightarrow{P}_{\mathcal{S}'}$ has $\mathcal{O}(p^3k)$ vertices, and thus by planarity of $\overrightarrow{P}_{\mathcal{S}'}$, the number of edges, and faces in $\overrightarrow{P}_{\mathcal{S}'}$ is also $\mathcal{O}(p^3k)$.

Definition 2. *Given a vertex $u \in V(G) \setminus M'$, we define the* hosted graph H_u *as a plane graph drawn within $\mathcal{D}_u$, with a (single) vertex in a face f of $\overrightarrow{P}_{\mathcal{S}'}$ if and only if $\mathcal{D}_u$ and f intersect, and with edges between vertices lying in adjacent faces of $\overrightarrow{P}_{\mathcal{S}'}$, if their common border intersects $\mathcal{D}_u$ (see Figure 4, right).*

Claim 3. For any inner-M'-vertex u, the hosted graph H_u is a tree.

Proof. By Properties (b) and (c), $\mathcal{D}_u$ does not contain a crossing, meaning that there is no v_1, v_2 such that $\mathcal{D}_u$ contains a point in $\partial\mathcal{D}_{v_1} \cap \partial\mathcal{D}_{v_2}$. Hence, the family $\{\mathcal{D}_v \cap \mathcal{D}_u, v \in N(u)\}$ is a laminar set family. This implies that H_u is a tree. ⌟

In order to count inner vertices, we distinguish two cases according to the maximum degree of H_u.

Claim 4. The number of vertices $u \in I_{M'}$ such that $\Delta(H_u) \geq 3$ is $\mathcal{O}(p^3k)$.

Proof. To see this, let us construct a bipartite pseudo-disk graph as follows. In one part, we consider the arcs of $\overrightarrow{P}_{\mathcal{S}'}$. These arcs are slightly shortened (so that they form an independent set) and slightly thickened (so that they intersect the adjacent regions we define below), see Figure 5. In the other part, we define a pseudo-disk $\mathcal{D}'_u$ for each vertex $u \in I_{M'}$ such that $\Delta(H_u) \geq 3$ as follows. Consider a vertex x of H_u such that $d_{H_u}(x) \geq 3$, and let $\mathcal{D}'_u$ be the intersection of $\mathcal{D}_u$ and the face of $\overrightarrow{P}_{\mathcal{S}'}$ containing x. Since triangle-free pseudo-disk graphs are planar, by [3, Lemma 9.24], the second part has size $\mathcal{O}(|E(\overrightarrow{P}_{\mathcal{S}'})|) = \mathcal{O}(p|M'|) = \mathcal{O}(p^3k)$. ⌟

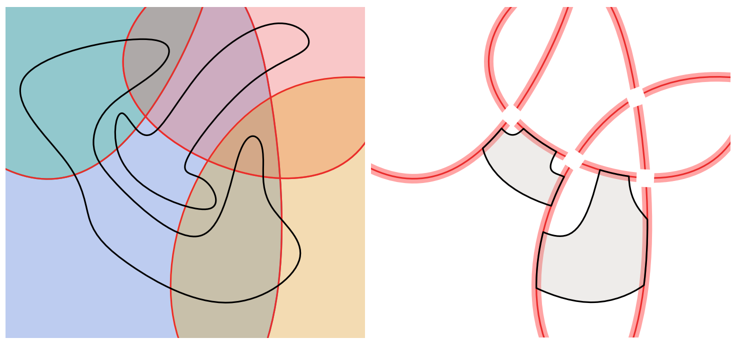

Fig. 5. Two inner-M' pseudo-disks with $\Delta(H_u) \geq 3$, and the bipartite pseudo-disk graph constructed to prove Claim 4.

Now, we focus on vertices $u \in I_{M'}$ such that $\Delta(H_u) \leq 2$. In that case, H_u is a path with possibly only one vertex, and as we show with bounded length.

Claim 5. For any vertex $u \in I_{M'}$ such that H_u is a path, this path has length at most $d(u)$, and $d(u) \leq 2p$. Furthermore, $N(u)$ can be split into two cliques of size at most p each.

Proof. For any neighbor v of u, $\mathcal{D}_v$ contains one of the two faces f_1 and f_2 of $\overrightarrow{P}_{\mathcal{S}'}$ containing the endpoints of the path H_u (see Figure 6 (left)). Otherwise, $\partial\mathcal{D}_u$ and $\partial\mathcal{D}_v$ would intersect in at least four points. Since the representation has ply at most p, there are at most $2p$ pseudo-disks containing one (or two) of these faces. Among those, the pseudo-disks containing f_1 clearly induce a clique, such as the remaining ones, since all of them contain f_2. ⌟

Even if these paths have bounded length, the number of such paths can still be arbitrarily large as we could draw them one "parallel" to each other, creating large class of twins (see Figure 6). Thus, we need the following kernelization rule for the feedback vertex set problem.

(twins) Consider an independent set S of $V(G) \setminus M'$ such that all the vertices in S have the same neighborhood N. If N admits a partition into two cliques and if $|S| > 4$, delete $|S| - 4$ vertices of S and keep the same parameter k.

Proof. (Safeness of the rule) Denote $v_1, \ldots, v_s$ the vertices in S, for some $s > 4$, and denote C_1, C_2 the two cliques partitioning N. Let $G' = G - \{v_5, \ldots, v_s\}$. Clearly G' cannot have a minimum feedback vertex set larger than G. Hence, it is sufficient to show that G' has a feedback vertex set X, that is also a feedback vertex set for G.

Consider any minimum feedback vertex set X of G', and let us transform it (if needed) into a minimum feedback vertex set intersecting N on at least $|N| - 1$ vertices. Observe first that for any i, $|C_i \setminus X| \leq 2$. If $|C_1 \setminus X| = 2$, then $\{v_1, v_2, v_3, v_4\} \subseteq X$, as otherwise the two vertices in $C_1 \setminus X$ and any v_i not in X would form a triangle in $G - X$. In such case, defining X' by replacing in X the vertices v_1, v_2, v_3, v_4 with the vertices in $C_1 \setminus X$ and in $C_2 \setminus X$ would result in the desired feedback vertex set. Indeed, the vertices v_1, v_2, v_3, v_4 are isolated in $G' - X'$, and $G' - \{v_1, v_2, v_3, v_4\} - X'$ is a forest, as $X \subseteq X' \cup \{v_1, v_2, v_3, v_4\}$.

Hence, we consider that $|C_1 \setminus X| \leq 1$ and $|C_2 \setminus X| \leq 1$. If $|C_1 \setminus X| = 1$ and $|C_2 \setminus X| = 1$, then $|\{v_1, v_2, v_3, v_4\} \cap X| \geq 3$, as otherwise the two vertices in $\{v_1, v_2, v_3, v_4\} \setminus X$, the vertex in $C_1 \setminus X$ and the vertex in $C_2 \setminus X$ would form a 4-cycle in $G - X$. In such case, defining X' by replacing in X two vertices in $\{v_1, v_2, v_3, v_4\} \cap X$ with the vertices in $C_1 \setminus X$ and in $C_2 \setminus X$ would result in the desired feedback vertex set. Indeed, the vertices v_1, v_2, v_3, v_4 not in X' are isolated in $G' - X'$, and $G' - \{v_1, v_2, v_3, v_4\} - X'$ is a forest, as $X \subseteq X' \cup \{v_1, v_2, v_3, v_4\}$.

We thus have a minimum feedback vertex set of G' intersecting N on at least $|N| - 1$ vertices. Since the vertices $v_5, \ldots, v_s$ are leaves or are isolated in $G - X$, we have that $G - X$ is the forest $G' - X$ with $s - 4$ leaves or isolated vertices added, and it is thus a forest. ⌟

Identifying the configuration required by the previous twins rule and updating the representation of G when we delete vertices can be done in polynomial time, implying that this kernelization can be performed in time polynomial in $|V(G)|$.

From now on, among the vertices $u \in I_{M'}$ such that H_u is a path, there are at most four vertices with the same neighborhood. We can thus focus on bounding the number of such vertices u with distinct neighborhoods. Indeed, if $I^*_{M'}$ is a maximal subset without twins (among vertices of $I_{M'}$ such that H_u is a path), then $|I_{M'}| \leq 4|I^*_{M'}| + \mathcal{O}(p^3 k)$ (by Claim 4). In the following subsection, we sketch how to bound $|I^*_{M'}|$ in order to obtain the following claim.

Claim 6. After the kernelization of Subsection 4.1, the number of inner-M'-vertices is $\mathcal{O}(p^5 |M'|)$, that is $|I_{M'}| = \mathcal{O}(p^7 k)$.

4.2 Sketch for bounding the size of $I^*_{M'}$

Given two hosted graphs H_u and H_v that are paths, we say that H_u and H_v are *identical* if their vertices are drawn in the same faces of $\overrightarrow{P}_{\mathcal{S}'}$, and if these faces are visited in the same order. Let $H^{\neq}_{M'}$ be a maximum set of pairwise non-identical paths H_u, for vertices $u \in I_{M'}$. As vertices v of $I^*_{M'}$ are non twins, their

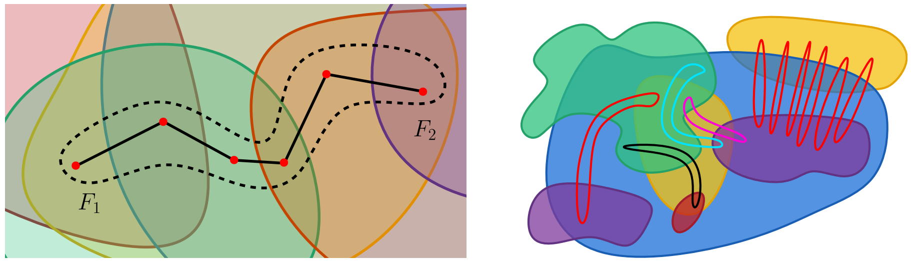

Fig. 6. Left: a hosted graph H_u that is a path. Right: some vertices $u \in I_{M'}$ such that H_u is a path. Some of these vertices have many twins.

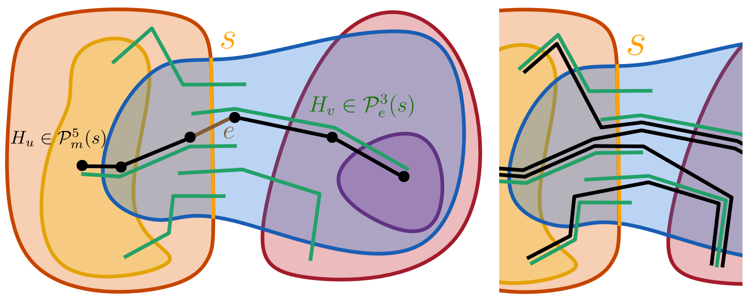

Fig. 7. In green the paths of $P_e^3(s)$. Left: in black and brown a path of $\mathcal{P}_m^5(s)$. Right: in black the paths of $P_m^5(s)$.

paths H_v are not identical, and thus $|I^*_{M'}| \leq |H^{\neq}_{M'}|$. Bounding the size of this set $H^{\neq}_{M'}$ thus bounds the sizes of $I^*_{M'}$ and $I_{M'}$. As a worst case, we assume that for any subpath P' of some path $P_1 \in H^{\neq}_{M'}$, there is a path $P_2 \in H^{\neq}_{M'}$ that is identical to P'.

We prove that $|H^{\neq}_{M'}| = \mathcal{O}(p^4 l)$, where l denotes the number of faces in $\overrightarrow{P}_{\mathcal{S}'}$, by showing that $|H^{\neq,i}_{M'}| = \mathcal{O}(i^3 l)$, where $H^{\neq,i}_{M'}$ is the set of paths $P \in H^{\neq}_{M'}$ of length i. Observe that $|H^{\neq,1}_{M'}| \leq |E(\overrightarrow{P}_{\mathcal{S}'})| = \mathcal{O}(l)$ as these paths cross a distinct arc of $\overrightarrow{P}_{\mathcal{S}'}$.

Due to lack of space, we now only sketch how to bound $|H^{\neq,i}_{M'}|$ for some odd $i \geq 3$. Let s be an edge of $\overrightarrow{P}_{\mathcal{S}'}$. Let $\mathcal{P}^i_m(s)$ be the set of paths of $H^{\neq,i}_{M'}$ whose middle edge e crosses s (see Figure 7 left), and $\mathcal{P}_e^{\lceil i/2 \rceil}(s)$ be the set of paths of $H^{\neq,\lceil i/2 \rceil}_{M'}$ whose first or last edge crosses s. The simple observation that any path from $\mathcal{P}^i_m(s)$ can be created by "gluing" two paths from $\mathcal{P}_e^{\lceil i/2 \rceil}(s)$ leads to the trivial bound $|\mathcal{P}^i_m(s)| \leq |\mathcal{P}_e^{\lceil i/2 \rceil}(s)|^2$. However, since these paths do not intersect each other, we can prove the key property that $|\mathcal{P}^i_m(s)| \leq 3|\mathcal{P}_e^{\lceil i/2 \rceil}(s)|$ (see Figure 7 right). By summing over all s, we get $|H^{\neq,i}_{M'}| \leq 6|H^{\neq,\lceil i/2 \rceil}_{M'}|$ (we get an extra factor 2 as each path in $H^{\neq,\lceil i/2 \rceil}_{M'}$ is counted twice, one time for each of its endpoint). This induction resolves in $|H^{\neq,i}_{M'}| \leq 6^{\log(i)}|H^{\neq,1}_{M'}| = \mathcal{O}(i^3 l)$.

5 Discussion

In this paper we gave algorithms for Feedback Vertex Set and Triangle Hitting in pseudo-disk graphs running in times $2^{\mathcal{O}(k^{9/10}\log k)}n^{\mathcal{O}(1)}$ and $2^{\mathcal{O}(k^{3/4}\log k)}n^{\mathcal{O}(1)}$, respectively. This generalizes the previous results [12] on disk graphs, with an improvement on the running time. On the other hand, as noted in the introduction, there is for both problems an ETH-based lower-bound of $2^{o(\sqrt{n})}$ [2]. So a natural problem is to get matching upper- and lower-bounds. We have no evidence to believe that our upper-bounds could be tight. Besides, our algorithms require a pseudo-disk representation of the input graph. So a second open problem is to provide a robust subexponential FPT algorithm for FVS in pseudo-disk graphs. The bulk of our algorithm for FVS consists in applying reduction rules to obtain an instance of size polynomial in the parameter. However this is not strictly speaking a kernel as we do not reduce the input instance but the subexponential number of instances produced by the preprocessing step. It could however be interesting to investigate if the ideas from our reduction steps could be useful for kernelization in pseudo-disk graphs.

References

1. An, S., Cho, K., Oh, E.: Faster algorithms for cycle hitting problems on disk graphs. In: Algorithms and Data Structures: 18th International Symposium, WADS 2023, Montreal, QC, Canada, July 31 – August 2, 2023, Proceedings. p. 29–42. Springer-Verlag, Berlin, Heidelberg (2023). https://doi.org/10.1007/978-3-031-38906-1_3, https://doi.org/10.1007/978-3-031-38906-1_3
2. Berthe, G., Bougeret, M., Gonçalves, D., Raymond, J.F.: Subexponential Algorithms in Geometric Graphs via the Subquadratic Grid Minor Property: The Role of Local Radius. In: Bodlaender, H.L. (ed.) 19th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2024). Leibniz International Proceedings in Informatics (LIPIcs), vol. 294, pp. 11:1–11:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2024). https://doi.org/10.4230/LIPIcs.SWAT.2024.11, https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.SWAT.2024.11
3. Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: Parameterized Algorithms. Springer Publishing Company, Incorporated, 1st edn. (2015)
4. Demaine, E.D., Fomin, F.V., Hajiaghayi, M., Thilikos, D.M.: Subexponential parameterized algorithms on bounded-genus graphs and h-minor-free graphs. J. ACM **52**(6), 866–893 (nov 2005). https://doi.org/10.1145/1101821.1101823, https://doi.org/10.1145/1101821.1101823
5. Diestel, R.: Graph theory 3rd ed. Graduate texts in mathematics **173**(33), 12 (2005)
6. Dvořák, Z., Lokshtanov, D., Panolan, F., Saurabh, S., Xue, J., Zehavi, M.: Efficient approximation for subgraph-hitting problems in sparse graphs and geometric intersection graphs. arXiv e-prints pp. arXiv–2304 (2023)
7. Dvořák, Z., Norin, S.: Treewidth of graphs with balanced separations. Journal of Combinatorial Theory, Series B **137**, 137–144 (2019). https://doi.org/https://doi.org/10.1016/j.jctb.2018.12.007

8. Fomin, F.V., Lokshtanov, D., Saurabh, S.: Excluded grid minors and efficient polynomial-time approximation schemes. Journal of the ACM (JACM) **65**(2), 1–44 (2018)
9. Fox, J., Pach, J.: Touching stings (2012), private communication
10. Kratochvíl, J.: Intersection graphs of noncrossing arc-connected sets in the plane. In: International Symposium on Graph Drawing. pp. 257–270. Springer (1996)
11. Lee, J.R.: Separators in Region Intersection Graphs. In: Papadimitriou, C.H. (ed.) 8th Innovations in Theoretical Computer Science Conference (ITCS 2017). Leibniz International Proceedings in Informatics (LIPIcs), vol. 67, pp. 1:1–1:8. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2017). `https://doi.org/10.4230/LIPIcs.ITCS.2017.1`
12. Lokshtanov, D., Panolan, F., Saurabh, S., Xue, J., Zehavi, M.: Subexponential parameterized algorithms on disk graphs (extended abstract). In: Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). pp. 2005–2031. SIAM (2022)