On Directed Feedback Vertex Set Parameterized by Treewidth

Marthe Bonamy[1], Łukasz Kowalik[2], Jesper Nederlof[3], Michał Pilipczuk[2], Arkadiusz Socała[2], and Marcin Wrochna[2](✉)

[1] CNRS, LaBRI, Talence, France
marthe.bonamy@labri.fr
[2] Institute of Informatics, University of Warsaw, Warsaw, Poland
{kowalik,michal.pilipczuk,as277575,m.wrochna}@mimuw.edu.pl
[3] Eindhoven University of Technology, Eindhoven, Netherlands
j.nederlof@tue.nl

Abstract. We study the Directed Feedback Vertex Set problem parameterized by the treewidth of the input graph. We prove that unless the Exponential Time Hypothesis fails, the problem cannot be solved in time $2^{o(t \log t)} \cdot n^{\mathcal{O}(1)}$ on general directed graphs, where t is the treewidth of the underlying undirected graph. This is matched by a dynamic programming algorithm with running time $2^{\mathcal{O}(t \log t)} \cdot n^{\mathcal{O}(1)}$. On the other hand, we show that if the input digraph is planar, then the running time can be improved to $2^{\mathcal{O}(t)} \cdot n^{\mathcal{O}(1)}$.

1 Introduction

In the Directed Feedback Vertex Set (DFVS) problem we are given a digraph G and the goal is to find a smallest *directed feedback vertex set* in it, that is, a subset X of vertices such that $G - X$ is acyclic. The arc-deletion version, Directed Feedback Arc Set (DFAS), differs in that the deletion set X has to consist of edges of G instead of vertices. The parameterized variants of these problems, where we ask about the existence of a solution of size at most k for a given parameter k, are arguably among central problems in the field of parameterized algorithms. Unfortunately, we are still far from a complete understanding of their complexity.

Work supported by the National Science Centre of Poland, grant number 2013/11/D/ST6/03073 (MP, MW). The work of Ł. Kowalik is a part of the project TOTAL that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 677651). This research is a part of projects that have received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreements No 714704 (AS). MP and MW are supported by the Foundation for Polish Science (FNP) via the START stipend programme. JN is supported by NWO Veni grant 639.021.438.

A. Brandstädt et al. (Eds.): WG 2018, LNCS 11159, pp. 65–78, 2018.
https://doi.org/10.1007/978-3-030-00256-5_6

Establishing the fixed-parameter tractability of DFVS was once a major open problem. It has been resolved by Chen et al. [2], who gave an algorithm for both DFVS and DFAS[1] with running time $2^{\mathcal{O}(k \log k)} \cdot n^{\mathcal{O}(1)}$, obtained by combining iterative compression with a smart application of important separators. Very recently, Lokshtanov et al. [15] revisited the algorithm of Chen et al. [2] and improved the running time to $2^{\mathcal{O}(k \log k)} \cdot (n+m)$; that is, the dependence on the size of the graph is reduced to linear, but the dependence on the parameter k is unchanged. Whether the running time can be improved to $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$, or even to $2^{o(k \log k)} \cdot n^{\mathcal{O}(1)}$, remains a challenging open problem [15]. We remark that the question of whether DFVS admits a polynomial kernel on general digraphs remains one of the central open problems in the field of kernelization.

A possible reason for why so little progress has been observed on such an important problem, is that the analysis of cut problems in directed graphs is far more complicated than in undirected graphs, and fewer basic techniques are available. For instance, consider the undirected counterpart of the problem, Feedback Vertex Set, where the goal is to delete at most k vertices from a given undirected graph in order to obtain a forest. While forests have a very simple combinatorial structure that can be exploited in many ways, acyclic digraphs form a much richer class that cannot be so easily captured. In particular, undirected graphs admitting a feedback vertex set of size k have treewidth at most $k+1$, and this tree-likeness of positive instances of undirected FVS makes the problem amenable to a variety of techniques related to treewidth; other basic techniques like branching and kernelization are also applicable. Acyclic digraphs may have arbitrarily large treewidth, whereas directed analogues of treewidth offer almost no algorithmic tools useful for the design of FPT algorithms. Therefore, for the study of DFVS and other directed cut problems in the parameterized setting, we are so far left with important separators and a handful of other more involved techniques; cf. [3,4,12,13,18].

In planar digraphs, the complexity of DFAS changes completely. As shown by Lucchesi and Younger [17], it is actually polynomial-time solvable (see also a different presentation by Lovász [16]). More precisely, this is a consequence of the proof of the Lucchesi–Younger theorem [17], which states that in planar digraphs, the minimum size of a directed feedback arc set is equal to the maximum size of a packing of arc-disjoint cycles. The proof is constructive and can be turned into a polynomial-time algorithm that computes a minimum directed feedback arc set together with a maximum cycle packing; see [19] for details.

On the other hand, it is easy to see that DFVS remains NP-hard on planar digraphs, as there is a simple reduction from Vertex Cover on planar graphs to Directed Feedback Vertex Set on planar digraphs: just pick an arbitrary ordering of vertices, orient all edges from left to right (giving an acyclic orientation), and replace every edge uv with a directed triangle on u, v, and a fresh vertex. To the best of our knowledge, no algorithm for DFVS with running time

[1] In general digraphs, DFVS and DFAS are well-known to be reducible to each other; see [5, Proposition 8.42 and Exercise 8.16]. These reductions, however, do not preserve planarity of the digraph in question.

$2^{o(k \log k)} \cdot n^{\mathcal{O}(1)}$ is known even for planar digraphs, which means that so far we are not able to exploit the planarity constraint in any useful way.

Our Contribution. The goal of this paper is to improve our understanding of DFVS by studying the parameterization by the treewidth of the input directed graph[2], with a particular focus on the planar setting. We first show that a semi-standard dynamic programming approach yields an algorithm with running time $2^{\mathcal{O}(t \log t)} \cdot n^{\mathcal{O}(1)}$.

Theorem 1. *There is an algorithm that given a digraph G of treewidth t on n vertices, runs in time $2^{\mathcal{O}(t \log t)} \cdot n^{\mathcal{O}(1)}$ and determines the minimum size of a directed feedback vertex set and of a directed feedback arc set in G.*

For the proof of Theorem 1, we define the following dynamic programming table (here for DFVS). For a node x of a tree decomposition of G, let B_x be the associated bag and let G_x be the subgraph induced in G by vertices residing in B_x or below x in the decomposition. Then, for every subset X of B_x and every ordering σ of $B_x \setminus X$, we store the smallest size of a subset Y of $V(G_x) \setminus B_x$ such that $G_x - (X \cup Y)$ is acyclic and admits a topological ordering whose restriction to $B_x \setminus X$ is exactly σ. Dynamic programming algorithm for DFAS is defined similarly. While we believe that this simple formulation of dynamic programming for DFVS and DFAS on a tree decomposition should have been known, we did not find it in the literature and hence we include it in the full version [1].

Our next result states then that the running time of the algorithm of Theorem 1 is tight under the Exponential Time Hypothesis (ETH) (see the Preliminaries section for definitions).

Theorem 2. *Unless ETH fails, there is no algorithm that determines the minimum size of a directed feedback vertex set or of a directed feedback arc set in a given digraph in time $2^{o(t \log t)} \cdot n^{\mathcal{O}(1)}$, where t is the treewidth of the input graph and n is the number of its vertices.*

The proof of Theorem 2 uses the approach of Lokshtanov et al. [14] for proving slightly super-exponential lower bounds for the complexity of parameterized problems. More precisely, we give a parameterized reduction from the $k \times k$ Hitting Set with thin sets problem, for which a lower bound excluding running time $2^{o(k \log k)} \cdot n^{\mathcal{O}(1)}$ under ETH was given in [14]. As an intermediate step, we use problems asking for permutations that satisfy certain constraints; we remark that somewhat similar constraint satisfaction problems, though with different constraints, were previously studied by Kim and Gonçalves [11].

Finally, we move to the setting of planar graphs, where we prove that the running time can be improved to $2^{\mathcal{O}(t)} \cdot n^{\mathcal{O}(1)}$.

Theorem 3. *There is an algorithm that given a planar digraph G of treewidth t on n vertices, runs in time $2^{\mathcal{O}(t)} \cdot n^{\mathcal{O}(1)}$ and determines the minimum size of a directed feedback vertex set in G.*

[2] Throughout this paper, the treewidth of a directed graph is defined as the treewidth of its underlying undirected graph.

It is well known that the treewidth of a planar graph on n vertices is bounded by $\mathcal{O}(\sqrt{n})$; see e.g. [7]. This yields the following.

Corollary 1. *There is an algorithm that given a planar digraph G on n vertices, runs in time $2^{\mathcal{O}(\sqrt{n})}$ and determines the minimum size of a directed feedback vertex set in G.*

Note that the algorithm of Corollary 1 is tight under ETH, due to the aforementioned simple reduction from Vertex Cover to DFVS which preserves planarity. Since Vertex Cover on planar graphs cannot be solved in time $2^{o(\sqrt{n})}$ under ETH (see [5, Theorem 14.6]), the same lower bound carries over to DFVS on planar digraphs (implying also a tight lower bound of $2^{o(t)} \cdot n^{\mathcal{O}(1)}$ for the parameterization by treewidth on planar digraphs).

The proof of Theorem 3 is perhaps conceptually the most interesting part of this work. The basic idea is to use *sphere-cut decompositions* of plane graphs [6,20]. Namely, as observed by Dorn et al. [6], from the results of Seymour and Thomas [20] it follows that every plane graph admits an optimum-width branch decomposition that respects the plane embedding in the following sense: each subgraph corresponding to a subtree of the decomposition is embedded into a disk so that the interface of the subgraph—vertices adjacent to the remainder of the graph—are embedded on the boundary of the disk. Such a branch decomposition is called a *sphere-cut decomposition*. Since branchwidth is linearly related to treewidth, in the proof of Theorem 3 we may focus on branch decompositions instead of tree decompositions.

As shown by Dorn et al. [6], the topological properties of sphere-cut decomposition can be exploited algorithmically to bound the number of relevant states in dynamic programming. This idea is instantiated in the technique of *Catalan structures* where for some connectivity problems, like Hamiltonian Cycle, the fact that the solution cannot self-intersect in the plane leads to an improvement on the number of states from $2^{\mathcal{O}(b \log b)}$ to $2^{\mathcal{O}(b)}$; here, b is the width of the considered sphere-cut decomposition. However, in the case of DFVS we cannot use Catalan structures directly, since we are not building any connected structure whose plane embedding would impose useful constraints.

Our main contribution here is that nevertheless, an improved upper bound on the number of relevant states can be shown, with a conceptually new reasoning. Consider a directed graph G embedded into a disk Δ and a subset T of its vertices that are placed on the boundary of Δ. Let the *connectivity pattern* induced by G on T be the reachability relation in G restricted to T^2: (s,t) are in the connectivity pattern if and only if in G there is a path from s to t. The crucial combinatorial statement (see Theorem 5) is as follows: the number of different connectivity patterns on T that may be induced by different digraphs G embedded in Δ is bounded by $2^{\mathcal{O}(|T|)}$; note that the naive bound would be $2^{\mathcal{O}(|T|^2)}$. This directly provides the sought upper bound on the number of relevant states in dynamic programming on a sphere-cut decomposition, leading to the proof of Theorem 3. To prove this statement, we show that every realizable connectivity pattern can be encoded using a constant number of simpler relations, each forming a directed outerplanar graph on $|T|$ vertices; the number of different such digraphs is $2^{\mathcal{O}(|T|)}$. In the proof that such an encoding is possible we use the result of Gyárfás that circle graphs are χ-bounded [8,9].

Organization. In Sect. 2 we establish notation and recall known relevant results. Section 3 concerns the main ingredient of the proof of Theorem 3, namely the combinatorial upper bound on the number of different connectivity patterns induced by disk-embedded directed graphs. Section 4 contains the hardness reduction for Theorem 2. Due to space restrictions the proofs of Theorem 3 and 1 and some proofs from Sects. 3 and 4 are deferred to the full version of this paper [1]. In these sections, theorems with deferred proofs are marked with †.

2 Preliminaries

Let $[k] := \{1, 2, \ldots, k\}$, and use standard graph notation, see e.g. [5]. The clique number of graph G is denoted $\omega(G)$, the chromatic number $\chi(G)$.

Chords and Circle Graphs: A *chord* is an unordered pair of distinct points on a circle, called *endpoints* of the chord; one may think of it as a straight line segment between its endpoints. Two chords $\{a, a'\}, \{b, b'\}$ of a circle *cross* if their endpoints are all distinct and a, b, a', b' occur in this order on the circle (clockwise or counter-clockwise). Intuitively this corresponds to the straight line segments aa' and bb' intersecting inside the circle. A *circle graph* is a graph whose vertices correspond to chords of a circle so that two vertices are adjacent if and only if the corresponding chords cross. A *circle graph with directed chords* is a circle graph in which every chord is directed; that is, it is an ordered pair. A directed chord with *tail* a and *head* b is denoted by (a, b). Let T be a finite set of points on a circle and let $R \subseteq T^2$ be a set of chords (directed or undirected). A *crossing* is a pair of crossing chords in R. The circle graph *induced* by R is the one with R as the vertex set where two chords from R are adjacent if they cross.

As introduced by Gyárfás [10], a class $\mathcal{C}$ of graphs closed under induced subgraphs is *χ-bounded* if there exists a function $f : \mathbb{N} \to \mathbb{N}$ such that for every graph $G \in \mathcal{C}$ we have $\chi(G) \leq f(\omega(G))$. Gyárfás [8,9] proved the following.

Theorem 4 ([8,9]). *The class of circle graphs is χ-bounded.*

ETH: The Exponential Time Hypothesis (ETH) states that for some $c > 0$, there is no algorithm for 3SAT with running time $\mathcal{O}(2^{cn})$, where n is the number of variables of the input formula. ETH has served as a basic assumption for countless complexity lower bounds of computational problems. We refer to [5, Chap. 14] for a comprehensive overview of applications in parameterized complexity.

3 Connectivity Patterns

In this section we present the main combinatorial result leading to the proof of Theorem 3, which is a reduction of the number of relevant dynamic programming states in the planar setting. This is done by bounding the number of "connectivity patterns" that can be induced by directed graphs embedded in a disk.

Suppose T is a finite set. A *connectivity pattern* on T is any quasi-order on T, that is, a reflexive and transitive relation $P \subseteq T^2$. For a directed graph G and a vertex subset $T \subseteq V(G)$, we define the connectivity pattern *induced by G on T* to be the reachability relation on T in G: (s,t) is in the relation iff there is a path in G from s to t.

The main goal of this section is to prove a result that roughly states the following: for a directed graph G drawn in a closed disk, with T be the vertices lying the boundary of the disk, there are only $2^{\mathcal{O}(|T|)}$ different possibilities for the connectivity pattern that G may induce. See Theorem 5 for a formal statement. As mentioned in the introduction, this result will be our main tool for limiting the number of relevant states in dynamic programming for Directed Feedback Vertex Set on planar graphs. Note that in general directed graphs, the number of different connectivity patterns induced on a vertex subset T may be as large as $2^{\Theta(|T|^2)}$. For instance, any subset of pairs with tail in the first half of T and head in the second half already gives that many possibilities.

The idea of the proof is that such connectivity patterns induced by directed planar graphs embedded in a disk can be generated from simpler relations, which contain enough pairs to infer all the other ones from planarity. This is formalized in the following definition.

Definition 1. *For a set T of points on a circle and a relation $R \subseteq T^2$, define the connectivity pattern on T* generated by R, *denoted* $\mathsf{gen}(R)$*, as follows: a pair $(s,t) \in T^2$ is included in* $\mathsf{gen}(R)$ *if and only if for each partition of the circle into two disjoint arcs X_s, X_t such that $s \in X_s$ and $t \in X_t$, there exist $s' \in X_s$ and $t' \in X_t$ which satisfy $(s',t') \in R$.*

In the above definition, as well as throughout this whole section, arcs on a circle may be open or closed from either side, unless explicitly stated.

It is easy to check that $R \subseteq \mathsf{gen}(R)$ and $\mathsf{gen}(R)$ is indeed reflexive and transitive, for any $R \subseteq T^2$. Hence $\mathsf{gen}(R)$ also contains the reflexive transitive closure of R, but it may be much larger still. Furthermore, one can observe that $\mathsf{gen}(\mathsf{gen}(R)) = \mathsf{gen}(R)$, but we will not use this property. We now show that a connectivity pattern induced by a graph is generated by itself; the goal will be then to find simpler relations generating this pattern.

Lemma 1. *Let G be a planar digraph drawn in a disk Δ, T be a subset of vertices drawn on the boundary of Δ, and P be the connectivity pattern on T induced by G. Then* $\mathsf{gen}(P) = P$.

Proof. Let C be the boundary of Δ; we may assume that C is a circle. Clearly $P \subseteq \mathsf{gen}(P)$. Now assume that $(s,t) \in \mathsf{gen}(P)$, that is, for each partition of C

into two disjoint arcs X_s, X_t such that $s \in X_s$ and $t \in X_t$, there exist $s' \in X_s$ and $t' \in X_t$ which satisfy $(s', t') \in P$. We will show that $(s, t) \in P$.

Assume the contrary, that is, $(s, t) \notin P$. Define $T_s = \{r \in T\colon (s, r) \in P\}$, see Fig. 1. Let X_t be the largest arc on C that contains t and is disjoint from T_s; this is well-defined since $t \notin T_s$ and $s \in T_s$. Define $X_s = C \setminus X_t$, thus (X_s, X_t) is a partition of C into two disjoint arcs. Since $s \in T_s$, we have $s \notin X_t$ and thus $s \in X_s$. From our assumption that $(s, t) \in \mathsf{gen}(P)$, there exist $s' \in X_s$ and $t' \in X_t$ that satisfy $(s', t') \in P$.

We have two cases: either $s' \in T_s$ or $s' \notin T_s$. If $s' \in T_s$, then $(s, s') \in P$ and consequently $(s, t') \in P$, since P is transitive due to being the reachability relation induced by G. But then $t' \in T_s$ and hence $t' \notin X_t$, a contradiction. Now assume $s' \notin T_s$; in particular $s' \neq s$. Let us move along the circle from s to t such that on the way we meet the point s'. Because the arc X_t was chosen to be the largest possible, between s' and t we meet a point $r \in T_s$. The arc X_t is connected, so between s and r we cannot meet any point from the set X_t, in particular t'. That is, s, s', r, t' appear in this order on the circle (either clockwise or counterclockwise). Since $r \in T_s$, we have $(s, r) \in P$ and $(s', t') \in P$. Therefore, in G there are directed paths from s to r and from s' to t'. These two paths must intersect since they are drawn in a disk, which yields a path in G from s to t'. We conclude that $t' \in T_s$ and hence $t' \notin X_t$, a contradiction. □

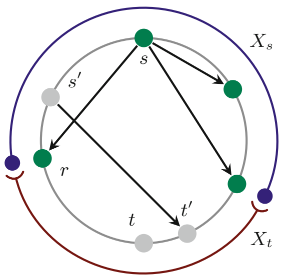

Fig. 1. Proof of Lemma 1: the induced pattern P shown as arrows, points in T_s depicted in green. (Color figure online)

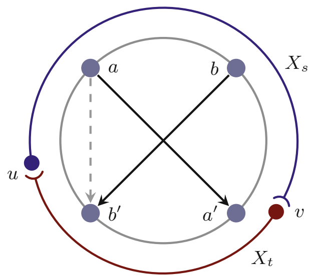

Fig. 2. Proof of Lemma 2.

The next lemma shows that generated connectivity patterns are closed under adding directed chords (a, b') whenever (a, a') and (b, b') cross. This operation (and its inverse) is the only one we will use to simplify the generating relation.

Lemma 2. *Let T be a finite set of points on a circle and let $R \subseteq T^2$. Let $a, b, a', b' \in T$ be distinct points that appear in this order on the circle, such that $(a, a') \in R$ and $(b, b') \in R$. Let $R' = R \cup \{(a, b')\}$. Then $\mathsf{gen}(R) = \mathsf{gen}(R')$.*

Proof. It is enough to prove that for each partition of the circle into two disjoint arcs X_s, X_t, the following two conditions are equivalent:

(1) There exist $s' \in X_s$ and $t' \in X_t$ which satisfy $(s', t') \in R$.
(2) There exist $s' \in X_s$ and $t' \in X_t$ which satisfy $(s', t') \in R'$.

Of course (1) implies (2). Now assume (2). If $(s', t') \in R$ the proof is finished, so suppose $(s', t') = (a, b')$. Let u, v be the ends of the arc X_s, see Fig. 2. We may assume without loss of generality that a, b, a', b' occur clockwise on the circle and are different from u, v; the latter is achieved by moving u, v slightly to points not belonging to T. Let $C_{a,b}$ be the arc of the circle from a (inclusive) to b (exclusive), going clockwise, and define $C_{b,a'}$, $C_{a',b'}$, $C_{b',a}$ analogously; these four arcs form a partition of the circle. Since $a \in X_s$ and $b' \in X_t$, we may assume that $u \in C_{b',a}$ and $v \notin C_{b',a}$. If $v \in C_{a,b}$ or $v \in C_{b,a'}$, then $a \in X_s$ and $a' \in X_t$ satisfy $(a, a') \in R$. Otherwise, if $v \in C_{a',b'}$, then $b \in X_s$ and $b' \in X_t$ satisfy $(b, b') \in R$. In both cases, (1) holds. □

Next, we prove that the generating relation can be simplified as long as it contains 4 pairwise crossing chords in the right order. The lemma after that shows how to obtain such chords from any set of 7 pairwise crossing chords.

Lemma 3 (†). *Let T be a finite set of points on a circle and let $R \subseteq T^2$. Let $a, b, c, d, x, y, z, u \in T$ be pairwise different points appearing in this order on the circle (clockwise or counterclockwise), such that $(a, x), (b, y), (c, z), (d, u) \in R$. Define*

$$R' = (R \setminus \{(b, y), (c, z)\}) \cup \{(b, z), (c, y)\}.$$

Then $\mathsf{gen}(R') = \mathsf{gen}(R)$ and the number of crossings in R' is smaller than in R.

Lemma 4 (†). *Suppose H is a circle graph with directed chords and $\omega(H) \geq 7$. Then there are distinct points a, b, c, d, x, y, z, u that appear in clockwise order on the circle such that $(a, x), (b, y), (c, z), (d, u)$ are pairwise crossing chords of H.*

Lemmas 3 and 4 allow us to conclude that any generating relation can be iteratively simplified until it contains no set of 7 pairwise crossing chords.

Lemma 5. *Let G be a planar graph drawn in a disk Δ, let T be a subset of vertices of G drawn on the boundary of Δ, and let $P \subseteq T^2$ be the connectivity pattern on T induced by G. Then there exists a relation $R \subseteq T^2$ such that $\mathsf{gen}(R) = P$ and the circle graph induced by R has clique number at most 6.*

Proof. By Lemma 1 there exists a relation $R \subseteq T^2$ (namely $R = P$) such that $\mathsf{gen}(R) = P$. Choose R such that $\mathsf{gen}(R) = P$ and the number of crossings in R is as small as possible. Without loss of generality assume that R does not contain pairs of the form (s, s) for $s \in T$, as such pairs may be removed without changing the generated relation; thus R is a set of directed chords with endpoints in T. Let ω be the clique number of the circle graph induced by R. If $\omega \leq 6$ we are done, so suppose $\omega \geq 7$. By Lemma 4, there are pairwise different points a, b, c, d, x, y, z, u that appear in clockwise order on the circle such that $(a, x), (b, y), (c, z), (d, u) \in R$. Define $R' = (R \setminus \{(b, y), (c, z)\}) \cup \{(b, z), (c, y)\}$. By Lemma 3, $\mathsf{gen}(R') = \mathsf{gen}(R) = P$ and R' has fewer crossings than R, a contradiction. □

Having obtained a generating relation with no large set of pairwise crossing chords, we will later partition it into a small number of sets of pairwise non-crossing chords using the χ-boundedness of circle graphs (Theorem 4). First, however, we bound the number of such non-crossing sets as follows.

Lemma 6 (†). *Let T be a finite set of points on a circle. Then every set of pairwise non-crossing chords with endpoints in T has at most $2|T| - 3$ chords, and there are $2^{\mathcal{O}(|T|)}$ different such sets.*

We are now ready to prove the main theorem of this section.

Theorem 5. *Let T be a set of n points on the boundary of a closed disk Δ. There exists a family $\mathcal{R}$ of relations $R \subseteq T^2$ such that $|\mathcal{R}| = 2^{\mathcal{O}(n)}$ and the following property is satisfied. For every planar digraph G drawn in Δ such that $T \subseteq V(G)$, the connectivity pattern induced by G on T is generated by some relation in $\mathcal{R}$.*

Proof. Denote by $\mathcal{R}$ the family of all sets of directed chords $R \subseteq T^2$ such that the circle graph induced by R has clique number at most 6. By Lemma 5 this family satisfies the claimed property and it remains to bound its size.

By χ-boundedness of circle graphs (Theorem 4), there exists a number $\chi_{\max}$ such that for $R \in \mathcal{R}$, the chromatic number of the circle graph induced by R is at most $\chi_{\max}$. The chords of any circle graph induced by some $R \in \mathcal{R}$ can thus be partitioned into $\chi_{\max}$ sets (possibly empty) such that no two chords in the same set cross. By Lemma 6, the number of possibilities to choose such a set of undirected, pairwise non-crossing chords is $2^{\mathcal{O}(n)}$, and any such set contains at most $2n - 3$ chords. Hence there are at most 2^{2n-3} possibilities to orient these chords. We conclude that indeed $|\mathcal{R}| \leq (2^{\mathcal{O}(n)} \cdot 2^{2n-3})^{\chi_{\max}} = 2^{\mathcal{O}(n)}$. □

With Theorem 5 in hand, the proof of Theorem 3 boils down to applying standard dynamic programming algorithm on a sphere-cut decomposition of the input graph. Each solved subproblem corresponds to a subgraph H of G embedded in a disk Δ, where each vertex of H that has a neighbor outside of H is embedded on the boundary of Δ; call the set of these vertices B. Then for each partition of B into X and T, and for each connectivity pattern P on T that can be induced by a digraph embedded in Δ, we compute that smallest size of a subset $Y \subseteq V(H) \setminus B$ such that $H - (X \cup Y)$ induces P on T; if there is no such subset, we store $+\infty$. It is straightforward to give recursive equations for this formulation. Moreover, Theorem 3 gives an upper bound of $2^{\mathcal{O}(t)}$ on the number of values computed for each H, where t is the treewidth, implying the running time of $2^{\mathcal{O}(t)} \cdot n^{\mathcal{O}(1)}$. Details, including an overview of sphere-cut decompositions, can be found in the full version [1].

4 Lower Bound

In this section we prove Theorem 2. The hardness reduction happens to work for both problems, producing exactly the same instances. We reduce from a problem shown hard by Lokshtanov et al. [14] (see also [5, Theorem 14.16]):

$k \times k$ Hitting Set with thin sets
Input: Family $\mathcal{F}$ of subsets of $[k] \times [k]$, each containing at most one element from each row
Question: Is there a set X containing exactly one vertex from each row of $[k] \times [k]$ such that $X \cap F \neq \emptyset$ for each $F \in \mathcal{F}$?

Theorem 6 ([14]). *Unless ETH fails,* $k \times k$ Hitting Set with thin sets *cannot be solved in time* $2^{o(k \log k)} \cdot n^{\mathcal{O}(1)}$*, where* n *is the number of input sets.*

We first define an intermediate problem. An *n-permutation d-constraint* is a tuple $(i_1, \dots, i_d) \in [n]^d$ of d different indices. A permutation $\sigma \colon [n] \to [n]$ *satisfies* such a constraint if $\sigma(i_1) < \sigma(i_2) < \cdots < \sigma(i_d)$. A *k-CNF n-permutation d-formula* is a conjunction of clauses, each of which is a disjunction of at most k n-permutation d-constraints. The *length* of a clause is the number of disjuncts (constraints) in it. Satisfaction of such a formula by a permutation $\sigma \colon [n] \to [n]$ is defined naturally.

We first show hardness for the satisfiability of 3-formulas, with the parameter k denoting both the length of clauses and the number of indices on which the permutation is defined.

Lemma 7. *Unless ETH fails, the satisfiability of a given k-CNF k-permutation 3-formula cannot be decided in time* $2^{o(k \log k)} \cdot n^{\mathcal{O}(1)}$*, where* n *is the formula size.*

Proof (sketch†). We only give the construction for the reduction, deferring the proof of its correctness to the appendix. Without loss of generality suppose $k \geq 3$. Let $\mathcal{F}$ be the input instance of $k \times k$ Hitting Set with thin sets. We construct in polynomial time a k-CNF $(2k+1)$-permutation 3-formula whose satisfiability is equivalent to the input instance $\mathcal{F}$, proving the claim by Theorem 6.

To an initially empty formula ϕ we add the following clauses, each with a single 3-constraint, to ensure that $\{k+1, \dots, 2k+1\}$ are ordered increasingly by the permutation:

$$(k+1, k+2, k+3),\ (k+2, k+3, k+4),\ \dots,\ (2k-1, 2k, 2k+1).$$

Then, for each $i \in [k]$ we add a clause with a single 3-constraint $(k+1, i, 2k+1)$. Finally, for each set $F \in \mathcal{F}$, we add the following clause C_F to ϕ: the clause C_F is the disjunction of constraints $(k+j, i, k+j+1)$ over all elements (i, j) of F. Since F contains at most one element of each row, the clause C_F is a disjunction of at most k constraints. □

Next, we show hardness for larger, but structured 2-formulas. For a 3-CNF n-permutation 2-formula ϕ, the *incidence graph* $I(\phi)$ of ϕ is the bipartite graph defined as follows: the vertex set is formed by indices from $[n]$ on one side and clauses of ϕ on the other side, and there is an edge between every clause and each index that occurs in some constraint of the clause. Thus, each clause has degree at most 6 in $I(\phi)$.

Lemma 8. *Unless ETH fails, the satisfiability of a given 3-CNF n-permutation 2-formula with incidence graph of treewidth t cannot be decided in time $2^{o(t \log t)} \cdot n^{\mathcal{O}(1)}$. This holds even for formulas in which every clause has length exactly 3 or 1, and has no repeating indices.*

Proof (sketch†). Let ϕ be a k-CNF k-permutation 3-formula. We will construct in polynomial time a 3-CNF n-permutation 2-formula ψ for some $n = \mathcal{O}(k^2)$ such that ψ is satisfiable iff ϕ is and the incidence graph of ψ has treewidth $\mathcal{O}(k)$. The claim then follows by Lemma 7.

The idea is that every 3-constraint (a, b, c) can be thought of as a conjunction $(a, b) \wedge (b, c)$ of two 2-constraints (expressing $\sigma(a) < \sigma(b) \;\wedge\; \sigma(b) < \sigma(c)$). Intuitively, we can then transform the obtained 'non-CNF formula' into a 3-CNF in a standard way: a clause $(x \wedge x') \vee (y \wedge y') \vee (z \wedge z') \vee \dots$ would be replaced by

$$\begin{aligned}&(p_1) \;\wedge\; (\neg p_1 \vee x \vee p_2) \;\wedge\; (\neg p_2 \vee y \vee p_3) \;\wedge\; (\neg p_3 \vee z \vee p_4) \;\wedge\; \dots \\ &\quad \wedge (\neg p_1 \vee x' \vee p_2) \;\wedge\; (\neg p_2 \vee y' \vee p_3) \;\wedge\; (\neg p_3 \vee z' \vee p_4) \;\wedge\; \dots \;\wedge\; (\neg p_n)\end{aligned}$$

where $p_1, p_2, p_3, \dots, p_n$ are fresh auxiliary variables not appearing anywhere else.

Formally, we will ask for n-permutations with $n := k + (2k+2)k$; the additional indices are in order to make room for 'auxiliary variables'. We construct ψ as an initially empty conjunction. Each clause C of ϕ is a disjunction $C_1 \vee \dots \vee C_{k'}$ $(k' \le k)$ of some 3-constraints $C_i = (a_i, b_i, c_i) \in [k]^3$. Let $j_1, j_2, \dots, j_{2k'+2} \in [n] \setminus [k]$ be some indices that were not yet used in any constructed clause. For each $i \in [k']$, we add the following clauses D_i and D'_i to ψ:

$$\begin{aligned}D_i &= (j_{2i}, j_{2i-1}) \vee (a_i, b_i) \vee (j_{2i+1}, j_{2i+2}) \\ D'_i &= (j_{2i}, j_{2i-1}) \vee (b_i, c_i) \vee (j_{2i+1}, j_{2i+2})\end{aligned}$$

We then add two clauses with a single constraint each: $Z = (j_1, j_2)$ and $Z' = (j_{2k'+2}, j_{2k'+1})$. Repeating this for each clause C of ϕ concludes the construction. Let $W(C)$ be the set consisting of clauses and indices used for C: clauses Z, Z', clauses D_i, D'_i for each $i \in [k']$, and indices $j_1, j_2, \dots, j_{2k'+2}$ as above. Then $[k]$ together with sets $W(C)$ for clauses C of ϕ form a partition of the vertex set of the incidence graph $I(\psi)$ of the constructed formula. Observe that if we remove all the k vertices corresponding to $[k]$, the only remaining edges in $I(\psi)$ have both endpoints within the same $W(C)$ for some clause C of ϕ, and each $W(C)$ has size at most $3k + 4$. This allows to bound the treewidth of $I(\psi)$ by $\mathcal{O}(k)$. Details and the correctness proof of the construction can be found in the appendix. □

We proceed to reducing the satisfiability problem for permutation formulas as described in Lemma 8 to Directed Feedback Vertex (Arc) Set. Permutations of $[n]$ will be encoded as orderings of a subset of n 'terminal' vertices in the constructed digraph, identified with indices from $[n]$. The digraph will contain gadgets ensuring that a permutation satisfies the original 3-CNF n-permutation

2-formula if and only if the corresponding ordering of terminals can be extended to a topological ordering of the whole digraph, after deleting a prescribed number of vertices (edges). The key element is the or-gadget depicted in Fig. 3, which encodes a clause that is a disjunction of three 2-constraints. Note that this or-gadget has 6 terminal vertices, named x_i, x'_i for $i \in [3]$. The final graph is obtained essentially by taking disjoint copies of the or-gadget and identifying their terminal vertices with terminals.

Lemma 9. *For an ordering $\prec$ of the terminal vertices of the or-gadget, $\prec$ can be extended to a topological ordering of the or-gadget with some 2 vertices (edges) deleted if and only if $x_1 \prec x'_1$ or $x_2 \prec x'_2$ or $x_3 \prec x'_3$. Furthermore, every subgraph of the or-gadget obtained by deleting at most one non-terminal vertex or an edge from it, contains a directed cycle.*

Proof. Given an ordering $\prec$ of the terminal vertices such that $x_1 \prec x'_1$, one can remove e_2 and e_3, or any two vertices incident to them, to create an acyclic subgraph of the or-gadget that admits a topological ordering extending $\prec$. The cases of orderings $\prec$ with $x_2 \prec x'_2$ and with $x_3 \prec x'_3$ are symmetric. Conversely, any removal of two vertices or edges from the or-gadget leaves some directed path $x_i \to x'_i$ ($i \in [3]$) unharmed, implying $x_i \prec x'_i$ in any topological ordering of the obtained subgraph. It is easy to check that two non-terminal vertices or edges of the or-gadget have to be removed to make it acyclic. □

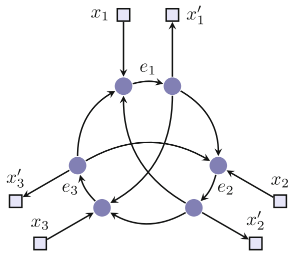

Fig. 3. The or-gadget, with terminal vertices marked as squares.

Proof (of Theorem 2, sketch†). We give a reduction from the satisfiability problem for 3-CNF n-permutation 2-formulas to DFVS and DFAS. More precisely, on the input of the reduction we are given a 3-CNF n-permutation 2-formula ψ with an incidence graph of treewidth t, where we assume that every clause of ψ has length exactly 3 or 1, and has no repeating indices. We will construct in polynomial time an equivalent instance of (the decision variant of) DFVS (DFAS) of treewidth $\mathcal{O}(t)$. This will prove the claim by Lemma 8.

We construct a digraph G starting from $[n]$ as the vertex set and no edges. For each clause of length 1 in ψ, let $(a, a') \in [n]^2$ be the unique constraint in it. Then we add an edge from a to a' to G. For each clause of length 3 in ψ, let $(a_1, a'_1), (a_2, a'_2), (a_3, a'_3) \in [n]^2$ be its constraints. Then we add a new copy of the or-gadget to G, and for each $i \in [3]$ we identify x_i and x'_i with a_i and a'_i, respectively. Finally, we set k, the target size of a directed feedback vertex (arc) set, to be twice the number of clauses of length 3 in ψ. The obtained instance (G, k) can be treated both as a DFVS instance and as a DFAS instance.

To bound the treewidth of G, observe that G can be obtained from $I(\psi)$ by replacing each vertex w corresponding to a clause with a copy of the or-gadget (if it represents a clause of length 3), or with just an edge between its original neighbors (if it represents a clause of length 1). □

5 Concluding Remarks

Our results do not provide any direct insight into the complexity of the classic parameterization: by the target solution size k. We hope, however, that the combinatorial tools we used in the proof of Theorem 3 may be useful for improving the running time for DFVS on planar digraphs, say to running time $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$, or for obtaining a somewhat incomparable running time $n^{\mathcal{O}(\sqrt{k})}$. Observe that there is a large gap between known results in this setting: while the classic reduction from Vertex Cover on planar graphs gives a lower bound excluding running time $2^{o(\sqrt{k})} \cdot n^{\mathcal{O}(1)}$ under ETH, no faster algorithm than $2^{\mathcal{O}(k \log k)} \cdot (n+m)$ from general digraphs [15] is known.

References

1. Bonamy, M., Kowalik, L., Nederlof, J., Pilipczuk, M., Socała, A., Wrochna, M.: On directed feedback vertex set parameterized by treewidth. arXiv abs/1707.01470 (2017)
2. Chen, J., Liu, Y., Lu, S., O'Sullivan, B., Razgon, I.: A fixed-parameter algorithm for the directed feedback vertex set problem. J. ACM **55**(5), 21:1–21:19 (2008)
3. Chitnis, R.H., Cygan, M., Hajiaghayi, M.T., Marx, D.: Directed subset feedback vertex set is fixed-parameter tractable. ACM Trans. Algorithms **11**(4), 28:1–28:28 (2015)
4. Chitnis, R.H., Hajiaghayi, M., Marx, D.: Fixed-parameter tractability of directed multiway cut parameterized by the size of the cutset. SIAM J. Comput. **42**(4), 1674–1696 (2013)
5. Cygan, M., et al.: Parameterized Algorithms. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21275-3
6. Dorn, F., Penninkx, E., Bodlaender, H.L., Fomin, F.V.: Efficient exact algorithms on planar graphs: exploiting sphere cut decompositions. Algorithmica **58**(3), 790–810 (2010)
7. Fomin, F.V., Thilikos, D.M.: New upper bounds on the decomposability of planar graphs. J. Graph Theory **51**(1), 53–81 (2006)
8. Gyárfás, A.: On the chromatic number of multiple interval graphs and overlap graphs. Discret. Math. **55**(2), 161–166 (1985)
9. Gyárfás, A.: Corrigendum: on the chromatic number of multiple interval graphs and overlap graphs. Discret. Math. **62**(3), 333 (1986)
10. Gyárfás, A.: Problems from the world surrounding perfect graphs. Applicationes Mathematicae **19**(3–4), 413–441 (1987)
11. Kim, E.J., Gonçalves, D.: On exact algorithms for the permutation CSP. Theor. Comput. Sci. **511**, 109–116 (2013)
12. Kratsch, S., Pilipczuk, M., Pilipczuk, M., Wahlström, M.: Fixed-parameter tractability of Multicut in directed acyclic graphs. SIAM J. Discret. Math. **29**(1), 122–144 (2015)
13. Kratsch, S., Wahlström, M.: Representative sets and irrelevant vertices: new tools for kernelization. In: FOCS 2012, pp. 450–459. IEEE Computer Society (2012)
14. Lokshtanov, D., Marx, D., Saurabh, S.: Slightly superexponential parameterized problems. In: SODA vol. 2011, pp. 760–776 (2011)

15. Lokshtanov, D., Ramanujan, M.S., Saurabh, S.: A linear time parameterized algorithm for directed feedback vertex set. CoRR abs/1609.04347 (2016)
16. Lovász, L.: On two minimax theorems in graph. J. Comb. Theory, Ser. B **21**(2), 96–103 (1976)
17. Lucchesi, C.L., Younger, D.H.: A minimax theorem for directed graphs. J. London Math. Soc **17**, 369–374 (1978)
18. Pilipczuk, M., Wahlström, M.: Directed multicut is $W[1]$-hard, even for four terminal pairs. In: SODA 2016, pp. 1167–1178. SIAM (2016)
19. Schrijver, A.: Combinatorial Optimization - Polyhedra and Efficiency. Springer, Heidelberg (2003)
20. Seymour, P.D., Thomas, R.: Call routing and the ratcatcher. Combinatorica **14**(2), 217–241 (1994)