How Much Does a Treedepth Modulator Help to Obtain Polynomial Kernels Beyond Sparse Graphs?*†

Marin Bougeret[1] and Ignasi Sau[2]

1 LIRMM, Montpellier, France
bougeret@lirmm.fr
2 CNRS, LIRMM, Montpellier, France, and
Departamento de Matemática, UFC, Fortaleza, Brazil
ignasi.sau@lirmm.fr

Abstract

In the last years, kernelization with structural parameters has been an active area of research within the field of parameterized complexity. As a relevant example, Gajarskỳ *et al.* [ESA 2013] proved that every graph problem satisfying a property called finite integer index admits a linear kernel on graphs of bounded expansion and an almost linear kernel on nowhere dense graphs, parameterized by the size of a c-treedepth modulator, which is a vertex set whose removal results in a graph of treedepth at most c for a fixed integer $c \geq 1$. The authors left as further research to investigate this parameter on general graphs, and in particular to find problems that, while admitting polynomial kernels on sparse graphs, behave differently on general graphs.

In this article we answer this question by finding two very natural such problems: we prove that Vertex Cover admits a polynomial kernel on general graphs for any integer $c \geq 1$, and that Dominating Set does *not* for any integer $c \geq 2$ even on degenerate graphs, unless NP $\subseteq$ coNP/poly. For the positive result, we build on the techniques of Jansen and Bodlaender [STACS 2011], and for the negative result we use a polynomial parameter transformation for $c \geq 3$ and an OR-cross-composition for $c = 2$. As existing results imply that Dominating Set admits a polynomial kernel on degenerate graphs for $c = 1$, our result provides a dichotomy about the existence of polynomial problems for Dominating Set on degenerate graphs with this parameter.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems, G.2.2 Graph Theory

Keywords and phrases parameterized complexity, polynomial kernels, structural parameters, treedepth, treewidth, sparse graphs

Digital Object Identifier 10.4230/LIPIcs.IPEC.2017.10

1 Introduction

Motivation. There is a whole area of parameterized algorithms and kernelization investigating the *complexity ecology* (see for example [18]), where the objective is to consider a *structural* parameter measuring how "complex" is the input, rather than the size of the solution. For instance, parameterizing a problem by the treewidth of its input graph has been a great success for FPT algorithms, triggered by Courcelle's theorem [4] stating that

* This work has been supported by project DEMOGRAPH (ANR-16-CE40-0028).
† A full version of this article is permanently available at https://arxiv.org/abs/1609.08095.

12th International Symposium on Parameterized and Exact Computation (IPEC 2017).
Editors: Daniel Lokshtanov and Naomi Nishimura; Article No. 10; pp. 10:1–10:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

any problem expressible in MSO logic is FPT parameterized by treewidth. However, the situation is not as good for kernelization, as many problems do not admit polynomial kernels when parameterized by treewidth unless NP $\subseteq$ coNP/poly [2].

Of fundamental importance within structural parameters are parameters measuring the so-called "distance from triviality" of the input graphs (a term that was first coined by Guo *et al.* [13]), like the size of a vertex cover (distance to an independent set) or of a feedback vertex set (distance to a forest). Unlike treewidth, these parameters may lead to both positive and negative results for polynomial kernelization. An elegant way to generalize these parameters is to consider a parameter allowing to *quantify* the triviality of the resulting instance, measured in terms of its treewidth. More precisely, for a positive integer c, a *c-treewidth modulator* of a graph G is a set of vertices X such that the treewidth of $G - X$ is at most c. Note that for $c = 0$ (resp. $c = 1$), a c-treewidth modulator corresponds to a vertex cover (resp. feedback vertex set).

Treewidth modulators have been extensively studied in kernelization, especially on classes of *sparse* graphs, where they have been at the heart of the recent developments of meta-theorems for obtaining linear and polynomial kernels on graphs on surfaces [3], minor-free graphs [8], and topological-minor-free graphs [12, 15], all based in a generic technique known as *protrusion replacement*. However, as observed in [11, 15], if one tries to move further in the families of sparse graphs by considering, for instance, graphs of bounded expansion, for several natural problems such as TREEWIDTH-t VERTEX DELETION (minimizing the number of vertices to be removed to get a graph of treewidth at most t), parameterizing by a treewidth modulator is as hard as on general graphs.

This observation led Gajarský *et al.* [11] to consider another type of modulators, namely *c-treedepth modulators* (defined analogously to c-treewidth modulators), where treedepth is a graph invariant – which we define in Section 2 – that plays a crucial structural role on graphs of bounded expansion and nowhere dense graphs [17]. Gajarský *et al.* [11] proved that any graph problem satisfying a property called *finite integer index* admits a linear kernel on graphs of bounded expansion and an almost linear kernel on nowhere dense graphs when parameterized by the size of a c-treedepth modulator. Shortly afterwards this result was obtained, the authors asked [5] to investigate this parameter on *general graphs*, namely to find natural problems that admit and that do not admit polynomial kernels parameterized by the size of a c-treedepth modulator. More precisely, are there natural problems Π_1 and Π_2 fitting into the framework of [11] such that Π_1/c-tdmod admits a polynomial kernel on general graphs, but Π_2/c-tdmod does not? (As defined in Section 2, "$/c$-tdmod" means "parameterized by the size of a c-treedepth modulator".)

Our results. In this article we answer the above question by proving that VERTEX COVER and DOMINATING SET are such problems Π_1 and Π_2, respectively. Let us now elaborate a bit more on our results, the techniques we use to prove them, and how do they compare to previous work in the area (see the preliminaries of Section 2 for any undefined terminology).

Note first that both VC/c-tdmod and DS/c-tdmod (where VC and DS stand for VERTEX COVER and DOMINATING SET, respectively) are FPT on general graphs, as they are FPT by treewidth [4], which is a smaller parameter than c-tdmod, as for any graph G and any integer $c \geq 0$, it holds that $\mathsf{tw}(G) \leq \mathsf{td}(G) - 1 \leq \text{c-tdmod}(G) + c - 1$. Thus, asking for polynomial kernels is a pertinent question.

In Section 3 we prove that VC/c-tdmod admits a polynomial kernel on general graphs. Our approach is based on the techniques introduced by Jansen and Bodlaender [14] to prove that VC/1-twmod (or equivalently, VC/FVS, where FVS stands for FEEDBACK VERTEX SET) admits a polynomial kernel. More precisely, we use three reduction rules inspired from the rules given in [14], and we present a recursive algorithm that, starting from a c-treedepth modulator, constructs an appropriate $(c-1)$-treedepth modulator and calls itself inductively. The kernel obtained in this manner has $x^{2^{\mathcal{O}(c^2)}}$ vertices, where x is the size of the c-treedepth modulator. This result completes the following panorama of structural parameterization for Vertex Cover, which has been a testbed for structural parameterizations in the last years:

- VC/1-twmod (or equivalently, VC/FVS) admits a polynomial kernel [14].
- VC/c-twmod for $c \geq 2$ does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly [6].
- VC/2-degmod (distance to a graph of maximum degree 2) and VC/c-CVD (distance to a disjoint collection of cliques of size at most c) admit a polynomial kernel [16]. Note that our result generalizes the latter kernel, as a disjoint collection of cliques of size at most c is a particular case of a graph having treedepth at most c.
- VC/pfm (distance to a pseudoforest, a graph in which every connected component has at most one cycle) admits a polynomial kernel [9].

In Section 4 we turn to negative results for Dominating Set. We provide a characterization, according to the value of c, of the existence of polynomial kernels for DS/c-tdmod on degenerate graphs. Indeed, using the results of Philip *et al.* [19] it is almost immediate to prove that DS/1-tdmod (or equivalently, DS/VC) admits a polynomial kernel on degenerate graphs. For $c \geq 3$, we rule out the existence of polynomial kernels for DS/c-tdmod on 2-degenerate graphs by a simple polynomial parameter transformation from DS/1-tdmod on general graphs, which does not admit polynomial kernels unless NP $\subseteq$ coNP/poly [7]. The remaining case, namely DS/2-tdmod, turns out to be more interesting, and we rule out the existence of polynomial kernels on 4-degenerate graphs by providing an or-cross-composition from 3-Sat. This dichotomy for the existence of polynomial kernels for DS/c-tdmod on degenerate graphs is to be compared with the dichotomy for VC/c-twmod on general graphs discussed above [14, 6].

As mentioned before, it is commonly admitted that almost no natural problem admits a polynomial kernel parameterized by tw, or even with td. However, to the best of our knowledge the only published negative results are those in [2], which together with [10] imply that IS/tw and DS/tw do not admit a polynomial kernel unless NP $\subseteq$ coNP/poly. As this result only holds for general graphs, for the sake of completeness we complete it in the full version, by showing that a large majority of the problems considered in [11] having an almost linear kernel parameterized by c-tdmod on nowhere dense graphs do not admit polynomial kernels parameterized by td, even on planar graphs of bounded maximum degree.

Due to space limitations, the proofs of the results marked with '($\star$)' have been moved to the full version. We also refer the reader to the full version for the definition and acronyms of problems considered in the paper.

2 Preliminaries

We present here just some preliminaries about graphs. The basic definitions about parameterized complexity can be found in the full version.

Unless explicitly mentioned, all graphs considered here are simple and undirected. Given a graph $G = (V, E)$ and $X \subseteq V$, we denote $N_X(v) = N(v) \cap X$, where $N(v) = \{u \in V \mid \{u, v\} \in E\}$. We denote by $\alpha(G)$ the size of a maximum independent set of G. For any function f defined on any induced subgraph of a given graph G, given a subset of vertices V' of G, we denote $f(V') = f(G[V'])$ (for example, $\alpha(V') = \alpha(G[V'])$). For any integer n, we denote $[n] = \{i \in \mathbb{N} \mid 1 \leq i \leq n\}$.

For the following definitions related to treedepth, bounded expansion, and nowhere dense graph classes, we refer the reader to [17] for more details, and we only recall here some basic notations and facts. The *treedepth* of a graph G (denoted $\mathsf{td}(G)$) is the minimum height of a rooted forest F (called a *treedepth decomposition*) such that G is a subgraph of the closure of F, where the closure of a rooted tree is the graph obtained by adding an edge between any internal vertex and all its ancestors, and the height of a rooted tree is the number of vertices in a longest path from the root to a leaf. Let $c \geq 1$ be an integer. A *c-treedepth modulator* is a subset of vertices $X \subseteq V$ such that $\mathsf{td}(G[V \setminus X]) \leq c$, and we denote by c-$\mathsf{tdmod}(G)$ the size of a smallest c-treedepth modulator of G. A c-treewidth modulator is defined in the same way. Recall that as these parameters are greater than their associated measure (i.e., $\mathsf{tw}(G) \leq c\text{-}\mathsf{twmod}(G) + c$) the negative results for kernelization by treewidth and treedepth do not immediately apply, but the positive FPT results do.

Concerning graph classes, we recall that in the sparse graph hierarchy, graphs of bounded expansion (BE) and nowhere dense graphs (ND) are related to classic sparse families as follows (see [17] for the definitions): planar graphs $\subseteq$ minor-free graphs $\subseteq$ BE $\subseteq$ ND. Note also that the class of graphs of bounded degeneracy is a natural superclass of BE (intuitively, BE also requires the shallow minors to be degenerate), and is incomparable with ND.

3 A polynomial kernel for VC/c-tdmod on general graphs

In this section we prove that for any positive integer c, VC/c-tdmod admits a polynomial kernel on general graphs. Recall that this was only known for VC/1-tdmod and VC/2-tdmod, as for $c = 1$ this corresponds to the standard parameterization and we can use the linear kernel of [1], and for $c = 2$ we have 1-twmod $\leq$ 2-tdmod (as a 1-twmod corresponds to the distance to a forest, while 2-tdmod corresponds to the distance to a star forest), and thus we can use the polynomial kernel of [14] for VC/1-twmod. We also recall that we cannot expect to extend our result to VC/c-twmod for any $c \geq 2$ [6].

As VC/c-tdmod and IS/c-tdmod are clearly equivalent for this parameterization, we provide the result for IS/c-tdmod. More specifically, in Subsection 3.1 we provide a polynomial kernel for a-c-tdmod-IS, an annotated version of our problem on hypergraphs defined below, and in Subsection 3.2 we derive a polynomial kernel for IS/c-tdmod.

3.1 A polynomial kernel for a-c-tdmod-IS/$(|X| + |\mathcal{H}|)$

Working with hypergraphs is useful because we will use a reduction rule identifying a subset X' of the modulator that cannot be entirely contained in a solution; this will be modeled by adding a hyperedge on the set X'.

> ANNOTATED c-TREEDEPTH MODULATOR INDEPENDENT SET (a-c-tdmod-IS)
>
> **Instance:** (G, X, k) where
> - $G = (V, E, \mathcal{H})$ is a hypergraph structured as follows: $V = X \uplus R$, $E = E_{X,R} \uplus E_{R,R}$ is a set of edges where edges in $E_{A,B}$ have one endpoint in A and the other in B, and $\mathcal{H} \subseteq 2^X$ is a set of hyperedges where each $H \in \mathcal{H}$ is entirely contained in X.
> - X is a c-treedepth modulator (as $G[V \setminus X]$ is no longer a hypergraph, its treedepth is correctly defined and we have $\mathsf{td}(V \setminus X) \leq c$).
> - k is a positive integer.
>
> **Question:** Decide whether $\alpha(G) \geq k$ (where an independent set in a hypergraph is a subset of vertices that does not contain any hyperedge, corresponding here to a subset $S \subseteq V$ such that for every $h \in E \cup H$, $h \nsubseteq S$).

Throughout this subsection $I = (G, X, k)$ denotes the input of a-c-tdmod-IS with $G = (V, E, \mathcal{H})$ and $V = X \uplus R$. Note that $G[X]$ is a hypergraph and that $G[R]$ is a graph, and that the parameter we consider here is $|X| + |\mathcal{H}|$. For any $X' \subseteq X$ and $R' \subseteq R$, observe that the notation $N_{R'}(X')$ is not ambiguous and denotes $\{v \in R' \mid \exists x \in X' \text{ with } \{x, v\} \in E\}$.

We use the following definition that was introduced in [14] for VC/1-twmod.

▶ **Definition 1 ([14]).** Given $X' \subseteq X$ and $R' \subseteq R$, let $\mathsf{conf}_{R'}(X') = \alpha(R') - \alpha(R' \setminus N_{R'}(X'))$ be the *conflicts* induced by X' on R'.

Intuitively, $\mathsf{conf}_{R'}(X')$ measures the loss in the size of a maximum independent set of R' due to X'. We extend the previous definition in the following way: for any $R' \subseteq R$ and any $Y' \subseteq R'$, let $\mathsf{conf}_{R'}(Y') = \alpha(R') - \alpha(R' \setminus Y')$. We can see that $\mathsf{conf}_{R'}(Y') = 0$ is equivalent to the existence of an independent set $S^* \subseteq R'$ such that $|S^*| = \alpha(R')$ and $S^* \cap Y' = \emptyset$.

▶ **Lemma 2.** *Let $R' \subseteq R$ be a connected component of R and let $Y' \subseteq R'$. If $\mathsf{conf}_{R'}(Y') > 0$, there exists $\bar{Y'} \subseteq Y'$ such that $\mathsf{conf}_{R'}(\bar{Y'}) > 0$ and $|\bar{Y'}| \leq f(c)$ with $f(c) = 2^c$.*

Proof. As it holds that $\mathsf{td}(R') \leq c$, let us consider a treedepth decomposition of R' with root r and $t \geq 1$ subtrees, where A_i, $i \in [t]$ is the vertex set of subtree i. We can partition $Y' = \bigcup_{i \in [t+1]} Y'_i$ with $Y'_i \subseteq A_i$ for $i \in [t]$, $Y'_{t+1} \subseteq \{r\}$, where the Y'_i's are possibly empty. We will prove the lemma by induction on c. Observe that $\sum_{i \in [t]} \alpha(A_i) \leq \alpha(R') \leq 1 + \sum_{i \in [t]} \alpha(A_i)$, and thus we distinguish two cases according to the value of $\alpha(R')$.

Case 1. $\alpha(R') = 1 + \sum_{i \in [t]} \alpha(A_i)$. In this case any maximum independent set S^* of R' contains r. Hence for every $i \in [t]$, $S^* \cap A_i$ is a maximum independent set in $A_i \setminus N_{A_i}(r)$, and thus $\alpha(A_i \setminus N_{A_i}(r)) = \alpha(A_i)$. Indeed, if we had $\alpha(A_i \setminus N_{A_i}(r)) < \alpha(A_i)$ for some i, then $|S^*|$ would be strictly smaller than $1 + \sum_{i \in [t]} \alpha(A_i)$.

If $r \in Y'$ (i.e., if $Y'_{t+1} \neq \emptyset$) then we can take $\bar{Y'} = \{r\}$ (as any optimal solution of R' must contain r we get $\alpha(R' \setminus \{r\}) < \alpha(R')$, and $|\bar{Y'}| = 1 \leq 2^c$), and thus we suppose henceforth that $Y'_{t+1} = \emptyset$.

We claim that there exists $i_0 \in [t]$ such that $\mathsf{conf}_{A_{i_0} \setminus N_{A_{i_0}}(r)}(Y'_{i_0}) > 0$. Indeed, otherwise we could define for any $i \in [t]$ an independent set $S_i \subseteq A_i \setminus N_{A_i}(r)$ with $|S_i| = \alpha(A_i \setminus N_{A_i}(r)) = \alpha(A_i)$ and $S_i \cap Y'_i = \emptyset$. Thus, $S^* = \{r\} \cup_{i \in [t]} S_i$ would be an independent set of size $\alpha(R')$, and as $Y'_{t+1} = \emptyset$ we would have $S^* \cap Y' = \emptyset$, a contradiction to the hypothesis that $\mathsf{conf}_{R'}(Y') > 0$. Thus, there exists $i_0 \in [t]$ such that $\mathsf{conf}_{A_{i_0} \setminus N_{A_{i_0}}(r)}(Y'_{i_0}) > 0$, and as $\mathsf{td}(A_{i_0} \setminus N_{A_{i_0}}(r)) < c$, by induction hypothesis there exists $\bar{Y'_{i_0}} \subseteq Y'_{i_0}$ such that $\mathsf{conf}_{A_{i_0} \setminus N_{A_{i_0}}(r)}(\bar{Y'_{i_0}}) > 0$ and $|\bar{Y'_{i_0}}| \leq 2^{c-1}$. Let us verify that $\bar{Y'} = \bar{Y'_{i_0}}$ satisfies $\mathsf{conf}_{R'}(\bar{Y'}) > 0$. Let S^* be an independent set of R' with $S^* \cap \bar{Y'} = \emptyset$. If $r \notin S^*$ then clearly $|S^*| < \alpha(R')$. Otherwise, $|S^*| = (\sum_{i \in [t]} |S^* \cap (A_i \setminus N_{A_i}(r))|) + 1 \leq \alpha(A_{i_0} \setminus N_{A_{i_0}}(r)) - 1 + (\sum_{i \in [t], i \neq i_0} \alpha(A_i \setminus N_{A_i}(r))) + 1 < \alpha(R')$.

Case 2. $\alpha(R') = \sum_{i \in [t]} \alpha(A_i)$. In this case there exists $i_0 \in [t]$ such that $\mathsf{conf}_{A_{i_0}}(Y'_{i_0}) > 0$. Indeed, otherwise we could define for any $i \in [t]$ an independent set $S_i \subseteq A_i$ with $|S_i| = \alpha(A_i)$ and $S_i \cap Y'_i = \emptyset$, and the existence of $S^* = \cup_{i \in [t]} S_i$ would be a contradiction to the hypothesis that $\mathsf{conf}_{R'}(Y') > 0$. Thus, by the induction hypothesis there exists $\bar{Y'_{i_0}} \subseteq Y'_{i_0}$ such that $\mathsf{conf}_{A_{i_0}}(\bar{Y'_{i_0}}) > 0$ and $|\bar{Y'_{i_0}}| \leq 2^{c-1}$.

If $r \in Y'$ (i.e., if $Y'_{t+1} \neq \emptyset$) then we can take $\bar{Y'} = \bar{Y'_{i_0}} \cup \{r\}$. Let us verify that $\mathsf{conf}_{R'}(\bar{Y'}) > 0$. Let S^* be an independent set of R' with $S^* \cap \bar{Y'} = \emptyset$. As S^* cannot contain r we have $|S^*| = \sum_{i \in [t]} |S^* \cap A_i| < \alpha(A_{i_0}) + \sum_{i \in [t], i \neq i_0} |S^* \cap A_i| = \alpha(R')$. Thus, we suppose from now on property $\mathbf{p_1} : Y'_{t+1} = \emptyset$.

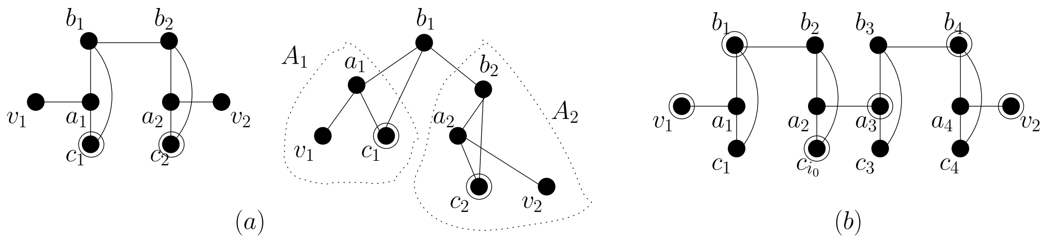

Figure 1 (a) Example of a graph $G[R']$ (left) with an associated treedepth decomposition (right) as used in Lemma 2, with $Y' = \{c_1, c_2\}$. This case corresponds to one of the subcases treated in Case 2 of Lemma 2, as $\alpha(R') = \alpha(A_1) + \alpha(A_2) = 4$, $\mathsf{conf}_{A_1}(Y_1') > 0$, $\mathsf{conf}_{A_2}(Y_2') = 0$. Moreover, $\mathbf{p_2}$ and $\mathbf{p_2'}$ are true, while $\mathbf{p_3}$ is false (but $\mathbf{p_3'}$ is true). (b) Example for $t = 2$ of the construction of Lemma 3, where the circled vertices belong to S.

Note that in this case (when $\mathbf{p_1}$ is true) we cannot simply set $\bar{Y'} = \bar{Y'_{i_0}}$, as shown in the example depicted in Figure 1. Indeed, in this example we would have $\bar{Y'} = \bar{Y'_{i_0}} = \{c_1\}$, however $\mathsf{conf}_{R'}(\{c_1\}) = 0$ as $S^* = \{b_1, v_1, c_2, v_2\}$ verifies $|S^*| = \alpha(R')$ and $S^* \cap \{c_1\} = \emptyset$.

Properties related to α. We claim that we can assume property $\mathbf{p_2}$: for every $i \neq i_0$, $\alpha(A_i \setminus N_{A_i}(r)) = \alpha(A_i)$. Indeed, if $\mathbf{p_2}$ is not true, then there exists $i_1 \neq i_0$, $i_1 \in [t]$ such that $\alpha(A_{i_1} \setminus N_{A_{i_1}}(r)) < \alpha(A_{i_1})$, and we set $\bar{Y'} = \bar{Y'_{i_0}}$. Let S^* be an independent set of R' with $S^* \cap \bar{Y'} = \emptyset$. If $r \notin S^*$ then as previously $|S^*| < \alpha(R')$, otherwise we get $|S^*| \leq \alpha(A_{i_0}) - 1 + \alpha(A_{i_1}) - 1 + (\sum_{i \in [t], i \neq i_0, i \neq i_1} \alpha(A_i)) + 1 < \alpha(R')$. Thus, we now assume $\mathbf{p_2}$.

Let us now prove the following property $\mathbf{p_2'}$: $\alpha(A_{i_0} \cup \{r\}) = \alpha(A_{i_0})$. By contradiction, suppose that there exists an independent set S_1^* of $A_{i_0} \cup \{r\}$ containing r such that $|S_1^*| = \alpha(A_{i_0}) + 1$. According to $\mathbf{p_2}$, for every $i \neq i_0$ there exists an independent set S_i of $A_i \setminus N_{A_i}(r)$ of size $\alpha(A_i)$, and thus $\alpha(R') > \sum_{i \in [t]} \alpha(A_i)$, a contradiction. Thus, we now assume $\mathbf{p_2'}$.

Properties related to $\mathsf{conf}_{A_i}(Y_i')$. Let us prove than we can assume the following property $\mathbf{p_3}$: for every $i \neq i_0, \mathsf{conf}_{A_i \setminus N_{A_i}(r)}(Y_i') = 0$. Indeed, if $\mathbf{p_3}$ is not true we can get the desired result as follows. Let $i_1 \neq i_0$, $i_1 \in [t]$ such that $\mathsf{conf}_{A_{i_1} \setminus N_{A_{i_1}}(r)}(Y_{i_1}') > 0$. We use the same arguments as in the previous paragraph and define $\bar{Y'} = \bar{Y'_{i_0}} \cup \bar{Y'_{i_1}}$. Note that $|\bar{Y'}| \leq |\bar{Y'_{i_0}}| + |\bar{Y'_{i_1}}| \leq 2^c$. Using the same notation, if $r \notin S^*$ then $|S^*| = (\sum_{i \in [t]} |S^* \cap A_i|) \leq \alpha(A_{i_0}) - 1 + (\sum_{i \in [t], i \neq i_0} \alpha(A_i)) < \alpha(R')$, and otherwise $|S^*| = (\sum_{i \in [t]} |S^* \cap (A_i \setminus N_{A_i}(r))|) + 1 \leq \alpha(A_{i_0}) - 1 + \alpha(A_{i_1}) - 1 + (\sum_{i \in [t], i \neq i_0, i \neq i_1} \alpha(A_i)) + 1 < \alpha(R')$. Thus, we now assume $\mathbf{p_3}$. Note that $\mathbf{p_2}$ and $\mathbf{p_3}$ imply property $\mathbf{p_3'}$: for every $i \neq i_0, \mathsf{conf}_{A_i}(Y_i') = 0$.

Case 2a. $\nexists S^*$ maximum independent set of R' such that $r \in S^*$. In this case, we set $\bar{Y'} = \bar{Y'_{i_0}}$. Let us prove that $\mathsf{conf}_{R'}(\bar{Y'}) > 0$. Let S^* be a maximum independent set of R' with $S^* \cap \bar{Y'} = \emptyset$. As $r \notin S^*$, we get $|S^*| = \sum_{i \in [t]} |S^* \cap A_i| \leq \alpha(A_{i_0}) - 1 + \sum_{i \in [t], i \neq i_0} \alpha(A_i) < \alpha(R')$.

Case 2b. $\exists S^*$ maximum independent set of R' such that $r \in S^*$. This implies that $\alpha(A_{i_0} \setminus N_{A_{i_0}}(r)) = \alpha(A_{i_0}) - 1$. Let us prove that $\mathsf{conf}_{A_{i_0} \setminus N_{A_{i_0}}(r)}(Y_{i_0}') > 0$. If it was not the case, there would exist an independent set $S_{i_0}^*$ of $A_{i_0} \setminus N_{A_{i_0}}(r)$ of size $\alpha(A_{i_0} \setminus N_{A_{i_0}}(r)) = \alpha(A_{i_0}) - 1$ such that $S_{i_0}^* \cap Y_{i_0}' = \emptyset$. By $\mathbf{p_3}$, there would exist, for every $i \neq i_0$, an independent set S_i^* of $A_i \setminus N_{A_i}(r)$ of size $\alpha(A_i \setminus N_{A_i}(r)) = \alpha(A_i)$ (by $\mathbf{p_2}$) such that $S_i^* \cap Y_i' = \emptyset$. Thus, $S^* = \{r\} \cup (\bigcup_{i \in [t]} S_i^*)$ would be an independent set of size $\alpha(R')$ such that $S^* \cap Y' = \emptyset$ (recall that by $\mathbf{p_1}$, $r \notin Y'$), a contradiction. Thus, we know that both $\mathsf{conf}_{A_{i_0} \setminus N_{A_{i_0}}(r)}(Y_{i_0}') > 0$

and $\mathsf{conf}_{A_{i_0}}(Y'_{i_0}) > 0$ (which was established at the beginning of Case 2). Using twice the induction hypothesis we get that there exists $\bar{Y'_{i_0}}^1 \subseteq Y'_{i_0}$ such that $\mathsf{conf}_{A_{i_0} \setminus N_{A_{i_0}}(r)}(\bar{Y'_{i_0}}^1) > 0$ and there exists $\bar{Y'_{i_0}}^2 \subseteq Y'_{i_0}$ such that $\mathsf{conf}_{A_{i_0}}(\bar{Y'_{i_0}}^2) > 0$, with both $|\bar{Y'_{i_0}}^1|$ and $|\bar{Y'_{i_0}}^2|$ bounded by 2^{c-1}. Thus, we set $\bar{Y'} = \bar{Y'_{i_0}}^1 \cup \bar{Y'_{i_0}}^2$. Let us verify that $\mathsf{conf}_{R'}(\bar{Y'}) > 0$. Let S^* be an independent set of R' with $S^* \cap \bar{Y'} = \emptyset$. If $r \in S^*$, then $|S^*| = \sum_{i\in[t]} |S^* \cap (A_i \setminus N_{A_i}(r))| + 1 = \alpha(A_{i_0} \setminus N_{A_{i_0}}(r)) - 1 + \sum_{i\in[t], i\neq i_0} \alpha(A_i) + 1 = \alpha(A_{i_0}) - 2 + \sum_{i\in[t], i\neq i_0} \alpha(A_i) + 1 < \alpha(R')$. Otherwise, $|S^*| = \sum_{i\in[t]} |S^* \cap A_i| = \alpha(A_{i_0}) - 1 + \sum_{i\in[t], i\neq i_0} \alpha(A_i) < \alpha(R')$. ◀

A first lower bound on the function f of Lemma 2 can be obtained by considering a clique R' on c vertices (hence, with $\mathsf{td}(R') = c$) and $Y' = R'$, as any $\bar{Y'} \subsetneq Y'$ satisfies $\mathsf{conf}_{R'}(\bar{Y'}) = 0$. However, as shown in Lemma 3 below, we can even obtain an exponential lower bound, showing that the function $f(c) = 2^c$ of Lemma 2 is almost tight.

▶ **Lemma 3** (⋆). *There exists a constant λ such that for any $c \geq \lambda$ there exists a graph $G = (R, E)$ and $Y \subseteq R$ such that $\mathsf{td}(G) = c$, $|Y| \geq 2^{c-3}$, $\mathsf{conf}_R(Y) > 0$, and for every $\bar{Y} \subsetneq Y$, $\mathsf{conf}_R(\bar{Y}) = 0$.*

▶ Remark. Lemma 2 was proven in [14] when R' is a forest and with $|\bar{Y'}| \leq 2$. Even if we already know that IS/2-twmod does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly [6], it remains interesting to observe that, in particular, this lemma becomes false for 2-twmod, as the graph of Lemma 3 has treewidth 2. This points out one crucial difference between c-treewidth and c-treedepth modulators.

Let us now start the description of the kernel for a-c-tdmod-IS/$(|X| + |\mathcal{H}|)$. Given an input (G, X, k) of a-c-tdmod-IS, we define the following three rules. Note that these rules and definitions (and the associated safeness proofs) correspond to Rules 1, 2, and 3 of [14], except that we now bound the sizes of the subsets by a function $f(c)$ instead of by 2.

▶ **Definition 4.** Given an input (G, X, k) of a-c-tdmod-IS (with $\mathsf{td}(G[R]) \leq c$ where $R = V \setminus X$), the *chunks* of the input are defined by $\mathcal{X} = \{X' \subseteq X \mid$ there is no $H \in \mathcal{H}$ such that $H \subseteq X'$, and $0 < |X'| \leq f(c)\}$, where $f(c) = 2^c$.

Intuitively, the chunks correspond to all possible small traces of an independent set of G in X. We are now ready to define the first two rules.

Reduction Rule 1: If there exists $u \in X$ such that $\mathsf{conf}_R(\{u\}) > |X|$, remove u from X.

Reduction Rule 2: If there exists $X' \in \mathcal{X}$ such that $\mathsf{conf}_R(X') > |X|$, add X' to $\mathcal{H}$.

▶ **Lemma 5** (⋆). *Rule 1 and Rule 2 are safe: if $I = (G, X, k)$ is the original input of a-c-tdmod-IS and $I^1 = (G^1, X^1, k)$ is the input after the application of Rule 1 or Rule 2, then I and I^1 are equivalent.*

Reduction Rule 3: If R contains a connected component R' such that for every $X' \in \mathcal{X}$, $\mathsf{conf}_{R'}(X') = 0$, delete R' from the graph and decrease k by $\alpha(R')$.

To prove that Rule 3 is safe we need the following lemma. Recall that we say that $X' \subseteq X$ is an independent set if and only if there is no $H \in \mathcal{H}$ such that $H \subseteq X'$.

▶ **Lemma 6** (⋆). *Let $I = (G, X, k)$ be an instance of a-c-tdmod-IS. Let R' be a connected component of R. If there exists an independent set $X' \subseteq X$ such that $\mathsf{conf}_{R'}(X') > 0$, then there exists $\bar{X'} \in \mathcal{X}$ such that $\mathsf{conf}_{R'}(\bar{X'}) > 0$.*

▶ **Lemma 7** ($\star$). *Rule 3 is safe: if $I = (G, X, k)$ is the original input of a-c-tdmod-IS and $I' = (G', X', k')$ is the input after the application of Rule 3, then I and I' are equivalent.*

▶ **Lemma 8** ($\star$). *Let $I = (G, X, k)$ be an instance of a-c-tdmod-IS, and let s be the number of connected components of $R = V \setminus X$. If none of Rule 1, Rule 2, or Rule 3 can be applied, then $s = \mathcal{O}(|X|^{f(c)+2})$, where f is the function of Lemma 2.*

We are now ready to present our polynomial kernel for a-c-tdmod-IS in Algorithm A below, which receives as input (I, c), where $I = (G, X, k)$ and X is a c-treedepth modulator.

$A(I, c)$:

1. If $c = 0$, return X. Otherwise:
2. While it is possible, apply Rule 1 (this rule suppresses vertices of X).
3. While it is possible, apply Rule 2 (this rule adds hyperedges of size at most $f(c)$ to $\mathcal{H}$).
4. Define the set $\mathcal{X}$, and while it is possible, apply Rule 3 (this rule suppresses some connected components of R and decreases k accordingly). Let $I_3 = (G_3, X_3, k_3)$ be the obtained instance, where $G_3 = (V_3, E_3)$ and $R_3 = V_3 \setminus X_3$.
5. For every connected component $R' \subseteq R_3$, compute an optimal treedepth decomposition of root $r_{R'}$. Let $X_r = \cup_{R' \subseteq R_3, R' \text{ connected}} \{r_{R'}\}$ be the set of roots.
6. Let $I' = (G' = (V', E', H'), X', k')$ be defined as follows. Let $V' = V_3$, $X' = X_3 \cup X_r$, and $Z = \{e \in E_3 \mid e \cap X_r \neq \emptyset \text{ and } e \cap X_3 \neq \emptyset\}$. Let $E' = E_3 \setminus Z, \mathcal{H}' = \mathcal{H}_3 \cup Z$ and $k' = k_3$ (I' corresponds to I_3 where we added X_r to the modulator, and consequently removed edges Z from E_3 and added them as hyperedges included in X'. Note that X' is now a $(c-1)$-treedepth modulator).
7. Return $A(I', c-1)$.

▶ **Theorem 9.** *For any fixed $c \geq 0$, Algorithm A is a polynomial kernel for a-c-tdmod-IS/$(|X| + |\mathcal{H}|)$. More precisely, for any input $I = (G, X, k)$ (with $G = (V, E, \mathcal{H})$, $R = V \setminus X$) where X is a c-treedepth modulator, Algorithm A produces an equivalent instance $\tilde{I} = (\tilde{G}, \tilde{X}, \tilde{k})$ (with $\tilde{G} = (\tilde{V}, \tilde{E}, \tilde{H})$, $\tilde{R} = \tilde{V} \setminus \tilde{X}$) where $|\tilde{X}| \leq \mathcal{O}(|X|^{2^{(c+1)(c+2)/2}})$, $|\tilde{\mathcal{H}}| \leq |\mathcal{H}| + \mathcal{O}(|X|^{2^{(c+1)(c+2)/2}})$, and $\tilde{R} = \emptyset$.*

Proof. Observe first that Algorithm A is polynomial for fixed c. Indeed, computing $\mathsf{conf}_{R'}(X')$ is polynomial (as $\mathsf{tw}(R') \leq \mathsf{td}(R')$ and it is well-known that IS/tw is FPT [4]) and there are at most $\mathcal{O}(|X|^c)$ applications of Rules 1 and 2, and $\mathcal{O}(s|X|^c)$ applications of Rule 3. Moreover, an optimal treedepth decomposition of each connected component can be computed in FPT time parameterized by c, using [17] or [20]. Let us prove the result by induction on c. The result is trivially true for $c = 0$. Let us suppose that the result holds for $c-1$ and prove it for c. Observe that X' is now a $(c-1)$-treedepth modulator, and thus we can apply the induction hypothesis on $A(I', c-1)$. For any $\ell \in [3]$, let $I_\ell = (G_\ell, X_\ell, k_\ell)$ with $G_\ell = (V_\ell, E_\ell, \mathcal{H}_\ell)$ and $R_\ell = V_\ell \setminus X_\ell$ denote the instance after exhaustive application of Rule ℓ, respectively.

Equivalence of the output. By Lemma 5 and Lemma 7, we know that Rules 1, 2, and 3 are safe, and thus that I and I_3 are equivalent. Note that I_3 is equivalent to I' as the underlying input is the same (except that some vertices were added to the modulator). As using induction hypothesis $A(I', c-1)$ outputs an instance $\tilde{I}$ equivalent to I', we get the desired result.

Size of the output. We have $|X_1| \leq |X|$, $|\mathcal{H}_1| = |\mathcal{H}|$, $|X_2| = |X_1|$, $|\mathcal{H}_2| \leq |\mathcal{H}_1| + |X_1|^{f(c)}$, $|X_3| = |X_2|$, $|\mathcal{H}_3| = |\mathcal{H}_2|$ (by Lemma 8, s, the number of connected components of R_3, verifies $s = \mathcal{O}(|X_3|^{f(c)+2})$), and $|X'| \leq |X_3| + s$, and $|\mathcal{H}'| \leq |\mathcal{H}_3| + s|X_3|$.

Thus we get $|X'| = \mathcal{O}(|X|^{f(c)+2}) = \mathcal{O}(|X|^{2^{c+1}})$ and $|\mathcal{H}'| = |\mathcal{H}| + \mathcal{O}(|X|^{f(c)+3})$. Using induction hypothesis we get that $|\tilde{X}| = \mathcal{O}(|X'|^{2^{c(c+1)/2}}) = \mathcal{O}(|X|^{2^{(c+1)(c+2)/2}})$, and that $|\tilde{\mathcal{H}}| = |\mathcal{H}'| + \mathcal{O}(|X'|^{2^{c(c+1)/2}}) = |\mathcal{H}| + \mathcal{O}(|X|^{2^c+3}) + \mathcal{O}(|X|^{2^{(c+1)(c+2)/2}}) = |H| + \mathcal{O}(|X|^{2^{(c+1)(c+2)/2}})$. ◀

3.2 Deducing a polynomial kernel for IS/c-tdmod

Observe first that we can suppose that the modulator is given in the input, i.e., that IS/c-tdmod $\leq_{\text{PPT}}$ c-tdmod-IS/$|X|$ ($\leq_{\text{PPT}}$ is defined in the full version). Indeed, given an input (G, x, k) of IS/c-tdmod (where x denotes the size of a c-treedepth modulator), using the 2^c-approximation algorithm of [11] for computing a c-treedepth modulator, wet get in polynomial time a set X such that $|X| \leq 2^c \cdot x$ and $\mathsf{td}(R) \leq c$, where $R = V \setminus X$.

Observe also that IS/$|X|$ $\leq_{\text{PPT}}$ a-c-tdmod-IS/$(|X| + |\mathcal{H}|)$ using the same set X and with $|\mathcal{H}| \leq |X|^2$. Now, as usual when using bikernels, we could claim that as IS is Karp NP-hard and as a-c-tdmod-IS is in NP, there exists a polynomial reduction from a-c-tdmod-IS, implying the existence of a polynomial kernel for IS/c-tdmod. However, let us make such a reduction explicit to provide an explicit bound on the size of the kernel.

▶ **Lemma 10** (⋆). *Let $I = (G, k)$ with $G = (X, \mathcal{H})$ be an instance of a-c-tdmod-IS as produced by Theorem 9 (as $R = \emptyset$ the set of vertices is reduced to X, and $\mathcal{H}$ is a set of hyperedges on X). We can build in polynomial time an equivalent instance $I' = (G', k')$ of* IS *with $G' = (V', E')$ where $|V'| \leq \mathcal{O}(|X| \cdot |\mathcal{H}|)$.*

Putting pieces together we immediately get the main theorem of this section.

▶ **Theorem 11.** *For every integer $c \geq 1$,* IS/c-tdmod *(or equivalently,* VC/c-tdmod*) admits a polynomial kernel on general graphs with $\mathcal{O}(x^{2^{\frac{1}{2}(c+1)(c+2)+1}})$ vertices, where x is the size of a c-treedepth modulator.*

4 Excluding polynomial kernels for DS/c-tdmod on degenerate graphs

Given a graph G, we define $G^{c\text{-sub}}$ as the graph obtained from G by subdividing each edge c times. In other words, we add a set $X_e = \{x_e^\ell \mid \ell \in [c]\}$ of c vertices of degree 2 for every edge $e \in E$ of G.

▶ **Observation 12** (⋆). *For any $c \geq 0$ and any $k \geq 0$, G has a dominating set of size k if and only if $G^{3c\text{-}sub}$ has a dominating set of size $k + mc$, where m is the number of edges of G.*

Let us start with the following proposition, which follows from existing negative results for Dominating Set parameterized by the size of a vertex cover [7].

▶ **Proposition 13** (⋆). DS/c-tdmod *does not admit a polynomial kernel on 2-degenerate graphs for any $c \geq 3$ unless $NP \subseteq coNP/poly$.*

▶ **Observation 14.** DS/1-tdmod *(or equivalently* DS/VC*) admits a polynomial kernel on degenerate graphs. Indeed, given an instance (G, k) of* DS/VC*, we compute in polynomial time a 2-approximate vertex cover X of G. If $|X| \leq k$ then we output a trivial* Yes*-instance, otherwise $VC(G) \geq \frac{k}{2}$ and we can apply the polynomial kernel for* DS/k *on degenerate graphs of Philip* et al. *[19].*

Thus, by Proposition 13 and Observation 14, the only remaining case for degenerate graphs is DS/2-tdmod. We would like to point out that the composition of [7] for DS/(k+VC)

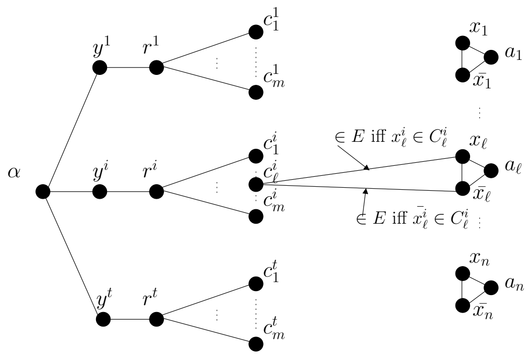

Figure 2 Example of the OR-cross-composition of Theorem 15.

on general graphs cannot be easily adapted to DS/2-tdmod on degenerate graphs, as for example subdividing each edge also leads to a result for DS/3-tdmod. Thus, we treat the case DS/2-tdmod on degenerate graphs using an ad-hoc reduction.

▶ **Theorem 15.** DS/2-tdmod *does not admit a polynomial kernel on* 4*-degenerate graphs unless* NP ⊆ coNP/poly*.*

Proof. We use an OR-cross-composition (see the full version for the definition) from 3-SAT. We consider t instances of 3-SAT, where for every $i \in [t]$, instance I^i has m_i clauses $\{C_j^i \mid j \in [m_i]\}$ and n_i variables $X^i = \{x_\ell^i \mid \ell \in [n_i]\}$, each clause containing three variables. We can safely assume that for every $i \in [t]$, we have $m_i = m$ and $n_i = n$.

Let us now construct a graph $G = (V, E)$ as follows; see Figure 2 for an illustration. We start by adding to V the set of vertices $\mathcal{X} = \bigcup_{\ell \in [n]} \{x_\ell, \bar{x}_\ell\}$ (and thus $|\mathcal{X}| = 2n$) and $C^i = \{c_\ell^i \mid \ell \in [m]\}$ for every $i \in [t]$. Let $C = \bigcup_{i \in [t]} C^i$. For every $i \in [t]$, $\ell \in [n]$, $j \in [m]$, we set $\{x_\ell, c_j^i\} \in E^i$ (resp. $\{\bar{x}_\ell, c_j^i\} \in E^i$) if and only if C_j^i contains x_ℓ^i (resp. $\bar{x}_\ell^i$). We add to E the set $\bigcup_{i \in [t]} E^i$. Then, we add to V the set $A = \{a_\ell \mid \ell \in [n]\}$, and create n triangles by adding to E edges $\{x_\ell, \bar{x}_\ell\}$, $\{a_\ell, x_\ell\}$, and $\{a_\ell, \bar{x}_\ell\}$ for every $\ell \in [n]$. Finally, we add to V the set $Y = \{y^i \mid i \in [t]\}$, $R = \{r^i \mid i \in [t]\}$, and a vertex α. Then, for every $i \in [t]$, we add to E edges $\{r^i, c_\ell^i\}$ for every $\ell \in [m]$, edges $\{r^i, y^i\}$, and edges $\{y^i, \alpha\}$. This concludes the construction of G. To summarize, G has $3n + t(m + 2) + 1$ vertices (vertices are partitioned into $V = (\mathcal{X} \cup A) \cup (C \cup Y \cup R) \cup \{\alpha\}$) and, in particular, for every $i \in [t]$, $G[\{r^i\} \cup C^i \cup y^i]$ is a star, and $G[\{\alpha\} \cup Y]$ is also a star.

The OR*-equivalence.* Let us prove that there exists $i \in [t]$ such that I^i is satisfiable if and only if G has a dominating set of size at most $k = n + t$. Suppose first, without loss of generality, that I^1 is satisfiable, and let $S_\mathcal{X} \subseteq \mathcal{X}$ be the set of n literals corresponding to this assignment (thus for every $\ell \in [n]$ we have $|S_\mathcal{X} \cap \{x_\ell, \bar{x}_\ell\}| = 1$). Let $S = S_\mathcal{X} \cup y^1 \cup (R \setminus \{r^1\})$. We have $|S| = n + t$, and S is a dominating set of G as

- $\mathcal{X} \cup A$ is dominated by $S_\mathcal{X}$,
- C^1 is dominated by $S_\mathcal{X}$ as it corresponds to an assignment satisfying I^1, and for every $i \in [t], i \geq 2$, C^i is dominated by r^i,
- $y^1 \in S$, and for every $i \in [t], i \geq 2$, y^i is dominated by r^i,
- r^1 is dominated by y^1, and for any $i \in [t], r \geq 2$, $r^i \in S$, and
- α is dominated by y^1.

For the other direction, let $S = S_1 \cup S_2$, with $S_1 = S \cap (\mathcal{X} \cup A)$, be a dominating set of G of size at most $k = n + t$. Without loss of generality, we can always suppose that $S_1 \subseteq \mathcal{X}$, as if $a_\ell \in S$ we can always remove a_ℓ from S and add (arbitrarily) x_ℓ or $\bar{x}_\ell$.

Let us first prove that $|S_1| = n$. Observe first that $|S_1| \geq n$ as dominating A requires at least n vertices. Suppose now by contradiction that $|S_1| > n$. Then, there would remain at most $t-1$ vertices to dominate R, which is not possible. Note that we even have that for any $\ell \in [n]$, $|S_1 \cap \{x_\ell, \bar{x_\ell}\}| = 1$, as every a^ℓ must be dominated and $|S_2| = t$.

Let us now analyze S_2 (recall that, by definition, $S_2 \subseteq (C \cup Y \cup R) \cup \{\alpha\}$). We cannot have that for every $i \in [t]$, $|S_2 \cap (C^i \cup r^i)| \geq 1$, as otherwise there would be no remaining vertex to dominate α. Thus, there exists i_0 such that $|S_2 \cap (C^{i_0} \cup r^{i_0})| = 0$. This implies that C^{i_0} is dominated by S_1. As for every $\ell \in [n]$, $|S_1 \cap \{x_\ell, \bar{x_\ell}\}| = 1$, S_1 corresponds to a valid truth assignment that satisfies all the C^i_ℓ's, $\ell \in [m]$, and the instance I^{i_0} is satisfiable.

Size of the parameter. Let $M = \mathcal{X} \cup A \cup \{\alpha\}$. As $G[V \setminus M]$ contains t disjoint stars, we have that 2-tdmod $(G) \leq |M| \leq \text{poly}(n)$, as required.

Degeneracy. Let us prove that G is 4-degenerate. Observe that any vertex in C has degree at most 4 (three neighbors in $\mathcal{X}$ and one in R). Thus, any ordering of $V(G)$ of the form $(C, R, Y, \alpha, \mathcal{X}, A)$ (with arbitrary order within each set) is a 4-elimination order of G. ◀

Note that for DS/c-tdmod with $c \geq 3$, the bound in the degeneracy given by Proposition 13 is best possible, as DS can be easily solved in polynomial time on 1-degenerate graphs, i.e., forests. On the other hand, for $c = 2$, in view of Theorem 15 only the existence of polynomial kernels for DS/2-tdmod on 2-degenerate and 3-degenerate graphs remains open.

5 Concluding remarks and further research

In this article we studied the existence of polynomial kernels for problems parameterized by the size of a c-treedepth modulator, on graphs that are not sparse. On the positive side, we proved that Vertex Cover (or equivalently, Independent Set) parameterized by the size x of a c-treedepth modulator admits a polynomial kernel on general graphs with $x^{2^{\mathcal{O}(c^2)}}$ vertices, for every $c \geq 1$. A natural direction is to improve the size of this kernel. Since Vertex Cover parameterized by the distance to a disjoint collection of cliques of size at most c does not admit a kernel with $\mathcal{O}(x^{c-\epsilon})$ vertices unless NP $\subseteq$ coNP/poly [16], and since a clique of size c has treedepth c, the same lower bound applies to our parameterization; in particular, this rules out the existence of a *uniform* kernel. However, there is still a large gap between both bounds, hence there should be some room for improvement.

On the negative side, we proved that Dominating Set parameterized by the size of a c-treedepth modulator does not admit a polynomial kernel on degenerate graphs for any $c \geq 2$. As Dominating Set with this parameterization admits a polynomial kernel on nowhere dense graphs [11], it follows that sparse graphs constitute the border for the existence of polynomial kernels. This leads us to the following natural question: are there smaller parameters for which Dominating Set still admits polynomial kernels on sparse graphs? Since considering as parameter the treedepth of the input graph does not allow for polynomial kernels (see the full version), we may consider as parameter the size x of a vertex set whose removal results in a graph of treedepth at most $b(x)$, for a function b that is not necessarily constant. We prove in the full version that Dominating Set does *not* admit polynomial kernels on graphs of bounded expansion for $b(x) = \Omega(\log x)$, unless NP $\subseteq$ coNP/poly. On the other hand, by combining the approach of Garnero *et al.* [12] to obtain explicit kernels via dynamic programming with the techniques of Gajarskỳ *et al.* [11] on graphs of bounded expansion, it can be shown – we omit the details here – that Dominating Set admits a polynomial kernel for $b(x) = \mathcal{O}(\log \log \log x)$ on graphs of bounded expansion whose expansion function f is not too "large" (that is, the function F that bounds the grad with rank d of the graphs in the family, see [17]), namely $f(d) = 2^{\mathcal{O}(d)}$. While this result is somehow anecdotal, we think that it may be the starting point for a systematic study of this topic.

References

1 Faisal N Abu-Khzam, Michael R Fellows, Michael A Langston, and W Henry Suters. Crown structures for vertex cover kernelization. *Theory of Computing Systems*, 41(3):411–430, 2007.

2 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.

3 Hans L. Bodlaender, Fedor V. Fomin, Daniel Lokshtanov, Eelko Penninkx, Saket Saurabh, and Dimitrios M. Thilikos. (Meta) Kernelization. In *Proc. of the 50th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 629–638. IEEE Computer Society, 2009.

4 B. Courcelle. The monadic second-order theory of graphs I: recognisable sets of finite graphs. *Information and Computation*, 85(12-75):663, 1990.

5 Marek Cygan, Lukasz Kowalik, and Marcin Pilipczuk. Open problem session of the Workshop on Kernelization (WorKer), Warsaw, Poland, 2013. Summary available at `http://worker2013.mimuw.edu.pl/slides/worker-opl.pdf`.

6 Marek Cygan, Daniel Lokshtanov, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. On the hardness of losing width. *Theory of Computing Systems*, 54(1):73–82, 2014.

7 Michael Dom, Daniel Lokshtanov, and Saket Saurabh. Kernelization lower bounds through colors and ids. *ACM Transactions on Algorithms*, 11(2):13, 2014.

8 Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Dimitrios M. Thilikos. Bidimensionality and kernels. In *Proc. of the 21st ACM-SIAM Symp. on Disc. Algorithms (SODA)*, pages 503–510, 2010.

9 Fedor V. Fomin and Torstein J. F. Strømme. Vertex cover structural parameterization revisited. *CoRR*, abs/1603.00770, 2016. `arXiv:1603.00770`.

10 L. Fortnow and R. Santhanam. Infeasibility of instance compression and succinct PCPs for NP. *Journal of Computer and System Sciences*, 77(1):91–106, 2011.

11 Jakub Gajarský, Petr Hlinený, Jan Obdrzálek, Sebastian Ordyniak, Felix Reidl, Peter Rossmanith, Fernando Sánchez Villaamil, and Somnath Sikdar. Kernelization using structural parameters on sparse graph classes. *Journal of Computer and System Sciences*, 84:219–242, 2017.

12 Valentin Garnero, Christophe Paul, Ignasi Sau, and Dimitrios M. Thilikos. Explicit linear kernels via dynamic programming. *SIAM Journal on Discrete Mathematics*, 29(4):1864–1894, 2015.

13 Jiong Guo, Falk Hüffner, and Rolf Niedermeier. A Structural View on Parameterizing Problems: Distance from Triviality. In *Proc. of the 1st International Workshop on Parameterized and Exact Computation (IWPEC)*, volume 3162 of *LNCS*, pages 162–173, 2004.

14 Bart Jansen and Hans Bodlaender. Vertex cover kernelization revisited: Upper and lower bounds for a refined parameter. In *Proc. of the 28th Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 9 of *LIPIcs*, pages 177–188, 2011.

15 Eun Jung Kim, Alexander Langer, Christophe Paul, Felix Reidl, Peter Rossmanith, Ignasi Sau, and Somnath Sikdar. Linear kernels and single-exponential algorithms via protrusion decompositions. *ACM Transactions on Algorithms*, 12(2):21, 2016.

16 Diptapriyo Majumdar, Venkatesh Raman, and Saket Saurabh. Kernels for structural parameterizations of vertex cover - case of small degree modulators. In *Proc. of the 10th International Symposium on Parameterized and Exact Computation (IPEC)*, volume 43 of *LIPIcs*, pages 331–342, 2015.

17 Jaroslav Nesetril and Patrice Ossona De Mendez. *Sparsity: Graphs, Structures, and Algorithms*. Algorithms and Combinatorics. Springer, 2012. URL: `https://hal.archives-ouvertes.fr/hal-00768681`.

18 Rolf Niedermeier. Reflections on multivariate algorithmics and problem parameterization. In *Proc. of the 27th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 5 of *LIPIcs*, pages 17–32, 2010.

19 Geevarghese Philip, Venkatesh Raman, and Somnath Sikdar. Solving dominating set in larger classes of graphs: FPT algorithms and polynomial kernels. In *Proc. of the 17th Annual European Symposium on Algorithms (ESA)*, volume 5757 of *LNCS*, pages 694–705, 2009.

20 Felix Reidl, Peter Rossmanith, Fernando Sánchez Villaamil, and Somnath Sikdar. A faster parameterized algorithm for treedepth. In *Proc. of the 41st International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 8572 of *Lecture Notes in Computer Science*, pages 931–942, 2014.