# Kernelization Using Structural Parameters on Sparse Graph Classes☆

Jakub Gajarský[a], Petr Hliněný[a], Jan Obdržálek[a], Sebastian Ordyniak[a], Felix Reidl[b], Peter Rossmanith[b], Fernando Sánchez Villaamil[b], Somnath Sikdar[b]

[a] *Faculty of Informatics, Masaryk University, Brno, Czech Republic,*
{gajarsky,hlineny,obdrzalek,ordyniak}@fi.muni.cz.
[b] *Theoretical Computer Science, Department of Computer Science, RWTH Aachen University, Aachen, Germany,*
{reidl,rossmani,fernando.sanchez,sikdar}@cs.rwth-aachen.de.

**Abstract**

We prove that graph problems with finite integer index have linear kernels on graphs of bounded expansion when parameterized by the size of a modulator to constant-treedepth graphs. For nowhere dense graph classes, our result yields almost-linear kernels. We also argue that such a linear kernelization result with a weaker parameter would fail to include some of the problems covered by our framework. We only require the problems to have FII on graphs of constant treedepth. This allows to prove linear kernels also for problems such as Longest-Path/Cycle, Exact-$s,t$-Path, Treewidth, and Pathwidth, which do not have FII on general graphs.

## 1. Introduction

Data preprocessing has always been a part of algorithm design. The last decade has seen steady progress in the area of *kernelization*, an area which deals with the design of polynomial-time preprocessing algorithms. These algorithms compress an input instance of a parameterized problem into an equivalent output instance whose size is bounded by some function of the parameter. Parameterized complexity theory guarantees the existence of such *kernels* for problems that are *fixed-parameter tractable*. Some problems admit stronger kernelization in the sense that the size of the output instance is bounded by a polynomial (or even linear) function of the parameter, the so-called *polynomial (or linear) kernels*.

Of great interest are *algorithmic meta-theorems*, results that focus on problem classes instead of single problems. In the area of graph algorithms, such

☆Research funded by DFG-Project RO 927/12-1 "Theoretical and Practical Aspects of Kernelization", the Czech Science Foundation under grant 14-03501S, and the European Social Fund and the state budget of the Czech Republic under project CZ.1.07/2.3.00/30.0009 (S. Ordyniak).

meta-theorems usually have the following form: all problems with a specific property admit, on a specific graph class, an algorithm of a specific type. We are specifically interested in meta-theorems that concern kernelization, for which a solid groundwork already exists. Before we delve into the history, we need to quickly establish the keystone property that drives all these meta-theorems: the notion of *finite integer index* (FII).

Roughly speaking, a graph problem has FII if there exists a finite set $\mathcal{S}$ of graphs such that every instance of the problem can be represented by a member of $\mathcal{S}$ alongside an integer "offset". This property is the basis of the *protrusion replacement rule* whereby protrusions (pieces of the input graph satisfying certain requirements) are replaced by members of the set $\mathcal{S}$. Finite integer index is an intrinsic property of the problem itself and is not directly related to whether it can be expressed in a certain logic. In particular, expressibility in the monadic second-order logic of graphs with vertices and edges ($\mathrm{MSO}_2$ and its extension to optimization problems abbreviated as $\mathrm{EMSO}_2$) does not imply FII (see [1] for sufficiency conditions for a problem expressible in counting MSO to have FII). As an example of this phenomenon, Hamiltonian Path has FII on general graphs whereas Longest Path does not, although both are $\mathrm{EMSO}_2$-expressible.

Now, the first steps towards a kernelization meta-theorem appeared in a paper by Guo and Niedermeier who provided a prescription of how to design linear kernels on planar graphs for graph problems which satisfy a certain distance property [2]. Their work built on the seminal paper by Alber, Fellows, and Niedermeier who showed that Dominating Set has a linear kernel on planar graphs [3]. This was followed by the first true meta-theorem in this area by Bodlaender et al. [1] who showed that graph problems that have FII and satisfy a property called *quasi-coverable*[1] admit linear kernels on bounded genus graphs. Shortly after [1] was published, Fomin et al. [4] proved a meta-theorem for linear kernels on $H$-minor-free graphs, a graph class that strictly contains graphs of bounded genus. A rough statement of their main result states that any graph problem that has FII, is *contraction bidimensional*, and satisfies a *separation property* has a linear kernel on graphs excluding a fixed graph as minor. This result was, in turn, generalized in [5] to $H$-topological-minor-free graphs, which strictly contain $H$-minor-free graphs. Here, the problems are required to have FII and to be *treewidth-bounding*: A graph problem is treewidth-bounding if yes-instances have a vertex set of size linear in the parameter, the deletion of which results in a graph of bounded treewidth. Such a vertex set is called a *modulator to bounded treewidth*. Prototypical problems that satisfy this condition are Feedback Vertex Set and Treewidth $t$-Vertex Deletion[2], when parameterized by the solution size.

We see that while these meta-theorems (viewed in chronological order) steadily covered larger graph classes, the set of problems captured in their framework diminished as the other precondition(s) became stricter. Surprisingly, this is not

[1]This property was called *quasi-compactness* in earlier version of [1]
[2]For problem definitions, see Appendix.

due to said preconditions: It turns out that they can be expressed in a unified manner and are therefore equally restrictive. The combined properties of bidimensionality and separability (used to prove the result on $H$-minor-free graphs) imply that the problem is treewidth-bounding (cf. Lemma 3.2 and 3.3 in [4]). Quasi-coverability on bounded genus graphs implies the same (cf. Lemma 6.4 in [1]). This demonstrates that all three previous meta-theorems on linear kernels implicitly or explicitly used treewidth-boundedness. Hence the diminishing set of problems can be blamed on the increasingly weaker interaction of the graph classes with the problem parameters, not the (only apparently) stricter precondition on the problems.

This insight motivates a different view on previous meta-theorems: problems that have FII admit linear kernels if parameterized by a *treewidth modulator* in classes excluding a topological minor. In small enough classes (bounded genus, apex-minor-free) the natural parameterization of problems satisfying some basic properties (quasi-coverable, contraction-bidimensionality) coincides with the parameterization by a treewidth-modulator. This change in perspective replaces the natural parameter—whose structural impact diminishes in larger sparse graph classes—by an explicit *structural* parameter which retains the crucial interaction between parameter and graph class. It also gives us, as we will see, the freedom to adapt the parameterization to our needs.

The next well-established level in the sparse-graph hierarchy [6] is formed by the classes of *bounded expansion*. The notion was introduced by Nešetřil and Ossona de Mendez [7] and subsumes graph classes excluding a fixed graph as a topological minor. It turns out that for these classes the serviceable parameterization by a treewidth modulator cannot work if we aim for linear kernels: Any graph class $\mathcal{G}$ can be transformed into a class $\tilde{\mathcal{G}}$ of bounded expansion by replacing every graph $G \in \mathcal{G}$ with $\tilde{G}$, obtained in turn by replacing each edge of $G$ by a path on $|V(G)|$ vertices. For problems like TREEWIDTH $t$-VERTEX DELETION and, in particular, FEEDBACK VERTEX SET this operation neither changes the instance membership nor does it increase the parameter. As both the problems do not admit kernels of size $O(k^{2-\epsilon})$ unless coNP $\subseteq$ NP/poly, by a result of Dell and Melkebeek [8], a linear kernelization result on bounded-expansion classes of graphs and under the treewidth-modulator parameterization would have to exclude both these natural problems.

In this work, we identify a structural parameter that indeed does allow linear kernels for all problems that have FII on graph classes of bounded expansion—the size of a tree*depth* modulator. This parameter not only increases under replacing edges with paths (a necessary prerequisite as we now know), but it also provides exactly the structure that seems necessary to obtain such a result. To put this parameterization into context, let us recap some previous work on structural parameters. Even outside the realm of sparse graphs, they have been used to zero in on those aspects of problems that make them intractable—a development that certainly fits the overall agenda of parameterized complexity. This research of alternative parameterizations has given rise to what is called the *parameterized ecology* [9].

Already the perhaps strongest structural parameter for graph-related problems—

the vertex cover number—makes up an interesting niche of said ecology, as we summarize now. Many problems that are W-hard or otherwise difficult to parameterize such as LONGEST PATH [10], CUTWIDTH [11], BANDWIDTH, IMBALANCE, DISTORTION [12], LIST COLORING, PRECOLORING EXTENSION, EQUITABLE COLORING, L($p$,1)-LABELING, and CHANNEL ASSIGNMENT [13] are (easily) fixed-parameter tractable (*fpt*) when parameterized by the vertex cover number. Some generalizations of vertex cover have also been successfully used as a parameter, e.g., [14, 15]. Even problems that do admit kernels in general or are fpt can benefit from such a strong structural parameter—for example, ODD CYCLE TRANSVERSAL (which admits a randomized and highly technical kernel), CHORDAL DELETION (which is fpt but does not admit a polynomial kernel), and $\mathcal{F}$-MINOR-FREE DELETION [16]. On the other hand, some problems are unlikely to admit polynomial kernelization even with this strong additional parameterization: DOMINATING SET, for example, has no polynomial kernel when parameterized by the solution size *and* the vertex cover number [17] and neither does the problem of finding a $t$-treewidth modulator parameterized by a $t'$-treewidth modulator (for most values of $t$ and $t'$) [18].

In light of previous work on structural parameters and the fact that a modulator to bounded treedepth is a significantly weaker parameter than the vertex cover number (which is the special case of a modulator to treedepth one), we conclude that treedepth modulator is a well-motivated choice in our case.

*Our contribution*

We show that, assuming FII, a parameterization by the size of a modulator to bounded treedepth allows for linear kernels in linear time on graph classes of bounded expansion. The same parameter yields almost-linear kernels on nowhere dense graph classes, which strictly contain those of bounded expansion. In particular, nowhere dense classes are the largest collections of graphs that may still be called sparse [6]. In these results we do not require a treedepth modulator to be supplied as part of the input, as we show that it can be approximated to within a constant factor.

Furthermore, we only need FII to hold on graphs of bounded treedepth, thus including problems which do not have FII in general. Some problems that are included because of this relaxation are LONGEST PATH/CYCLE, PATHWIDTH and TREEWIDTH, none of which have polynomial kernels with respect to their standard parameters, even on sparse graphs, since they admit simple AND/OR-Compositions [19]. Problems covered by our framework include also HAMILTONIAN PATH/CYCLE, several variants of DOMINATING SET, (CONNECTED) VERTEX COVER, CHORDAL VERTEX DELETION, FEEDBACK VERTEX SET, INDUCED MATCHING, BRANCHWIDTH and ODD CYCLE TRANSVERSAL. In particular, we cover all problems included in earlier frameworks [1, 4, 5]. We emphasize, however, that this paper does not subsume the former results since our parameter (the size of a modulator to constant tree*depth*) is necessarily larger or equal than the parameter of the previous results (the size of a modulator to constant tree*width*).

*Organization*
Our notation and the main definitions pertaining to graph classes can all be found in Section 2. Section 3 deals with the notion of finite integer index and the protrusion machinery. In Section 4, we prove our meta-theorems for graph classes of bounded expansion and for nowhere dense classes. In Section 5 we list problems already known to have finite integer index, and show that connectivity problems such as LONGEST PATH, and the width-measure problem BRANCHWIDTH have FII in appropriate graph classes. In Section 6 we show that the FII results can be extended to include the width-measure problems TREEWIDTH and PATHWIDTH, (the cases of which are surprisingly more difficult than that of branchwidth). These findings then enable us to apply our meta-theorem and obtain new kernelization results for the listed problems. We conclude in Section 7 with some open problems. In the appendix, we define (some of) the graph problems that we deal with in this paper.

## 2. Preliminaries

We use standard graph-theoretic notation (see [20] for any undefined terminology). All our graphs are finite and simple. Given a graph $G$, we use $V(G)$ and $E(G)$ to denote its vertex and edge sets. For convenience we assume that $V(G)$ is a totally ordered set, and use $uv$ instead of $\{u, v\}$ to denote the edges of $G$. By $H \subseteq G$ we mean that $H$ is a subgraph of $G$, and by $H \subseteq_{\text{ind}} G$ we denote that $H$ is an induced subgraph of $G$. For $X \subseteq V(G)$, we let $G[X]$ denote the subgraph of $G$ induced by $X$, and we define $G - X := G[V(G) \setminus X]$.

Since we will mainly be concerned with sparse graphs in this paper, we let $|G|$ denote the number of vertices in the graph $G$. The distance $d_G(v, w)$ of two vertices $v, w \in V(G)$ is the length (number of edges) of a shortest $v, w$-path in $G$ and $\infty$ if $v$ and $w$ lie in different connected components of $G$. The diameter $diam(G)$ of a graph is the length of a longest shortest path between all pairs of vertices in $G$. A complete subgraph of $G$ is called a *clique* and we denote by $\omega(G)$ the largest size of a clique of $G$.

The concept of neighborhood is used heavily throughout the paper. The neighborhood of a vertex $v \in V(G)$ is the set $N^G(v) = \{w \in V(G) \mid vw \in E(G)\}$, the degree of $v$ is $\deg^G(v) = |N^G(v)|$, and the closed neighborhood of $v$ is defined as $N^G[v] := N^G(v) \cup \{v\}$. We extend this naturally to sets of vertices and subgraphs: For $S \subseteq V(G)$ we denote $N^G(S)$ the set of vertices in $V(G) \setminus S$ that have at least one neighbor in $S$, and for a subgraph $H$ of $G$ we put $N^G(H) = N^G(V(H))$. Finally if $X$ is a subset of vertices disjoint from $S$, then $N^G_X(S)$ is the set $N^G(S) \cap X$ (and similarly for $N^G_X(H)$). Given a graph $G$ and a set $W \subseteq V(G)$, we also define $\partial^G(W)$ as the set of vertices in $W$ that have a neighbor in $V(G) \setminus W$. Note that $N^G(W) = \partial^G(V(G) \setminus W)$. A graph $G$ is $d$-degenerate for $d \in \mathbf{N_0}$ if every subgraph $G'$ of $G$ contains a vertex $v \in V(G')$ with $\deg^{G'}(v) \leqslant d$. The degeneracy of $G$ is the smallest $d$ such that $G$ is $d$-degenerate.

A set $S$ of vertices of a graph $G$ is a *separator* if $G - S$ is not connected. In particular, we say that $S$ *separates* two (not necessarily disjoint) sets $A$ and

$B$ of vertices of $G$ if $A \cap B \subseteq S$ and $G - S$ does not contain a path between a vertex in $A \setminus S$ and a vertex in $B \setminus S$. We say that a set of vertices $S$ is a *minimum* separator for $A$ and $B$ if there is no (cardinality-wise) smaller set of vertices separating $A$ and $B$ in $G$. Given a set $A$ of vertices we say that a vertex $v$ is *reachable* from $A$ in $G$ if $G$ contains a path between a vertex from $A$ and $v$.

In the rest of the paper we often drop the index $G$ from all the notation if it is clear which graph is being referred to.

### 2.1. Minors and shallow minors

We start by defining the notion of edge contraction. Given an edge $e = uv$ of a graph $G$, we let $G/e$ denote the graph obtained from $G$ by *contracting* the edge $e$, which amounts to deleting the endpoints of $e$, introducing a new vertex $w_{uv}$, and making it adjacent to all vertices in $(N^G(u) \cup N^G(v)) \setminus \{u, v\}$. By contracting $e = uv$ to the vertex $w$, we mean that the vertex $w_{uv}$ is renamed as $w$. *Subdividing* an edge is, in a sense, an opposite operation to contraction. A graph $G$ is called a $\leqslant k$*-subdivision* of a graph $H$ if (some) edges of $H$ are replaced by paths of length at most $k + 1$.

A *minor* of $G$ is a graph obtained from a subgraph of $G$ by contracting zero or more edges. In a more general view, if $H$ is isomorphic to a minor of $G$, then we call $H$ a minor of $G$ as well, and we write $H \preceq_m G$. A graph $G$ is $H$*-minor-free* if $H \not\preceq_m G$.

We next introduce the notion of a shallow minor.

**Definition 2.1** (Shallow minor [6])**.** For an integer $d$, a graph $H$ is a *shallow minor at depth* $d$ of $G$ if there exists a set of disjoint subsets $V_1, \ldots, V_p$ of $V(G)$ such that

1. each graph $G[V_i]$ has radius at most $d$, meaning that there exists $v_i \in V_i$ (a *center*) such that every vertex in $V_i$ is within distance at most $d$ in $G[V_i]$ from $v_i$; and
2. there is a bijection $\psi \colon V(H) \to \{V_1, \ldots, V_p\}$ such that for $u, v \in V(H)$, $uv \in E(H)$ only if there is an edge in $G$ with an endpoint each in $\psi(u)$ and $\psi(v)$.

The sets $V_1, \ldots, V_p$ are called the *branch sets* of this particular embedding of the minor. Note that if $u, v \in V(H)$ with branch sets $\psi(u) = V_i$ and $\psi(v) = V_j$, then the distance of the centers $v_i \in V_i$ and $v_j \in V_j$ is bounded by $d_G(v_i, v_j) \leqslant (2d + 1) \cdot d_H(u, v)$. The class of shallow minors of $G$ at depth $d$ is denoted by $G \triangledown d$. This notation is extended to graph classes $\mathcal{G}$ as well: $\mathcal{G} \triangledown d = \bigcup_{G \in \mathcal{G}} G \triangledown d$.

Note that, in particular, $G \triangledown 0$ is the class of all subgraphs of $G$, and $G \triangledown \infty$ is the class of all minors of $G$.

### 2.2. Parameterized problems, kernels and treewidth

In this paper we deal with parameterized problems where the value of the parameter is not explicitly specified in the input instance. This situation is slightly different from the usual case where the parameter is supplied with the input and a parameterized problem is defined as sets of tuples $(x, k)$ as in [21]. As such, we find it convenient to adopt the definition of Flum and Grohe [22] and we feel that this is the approach one might have to choose when dealing with generalized parameters as is done in this paper.

Let $\Sigma$ be a finite alphabet. A parameterization of $\Sigma^*$ is a mapping $\kappa\colon \Sigma^* \to \mathbf{N_0}$ that is polynomial-time computable. A parameterized problem $\Pi$ is a pair $(Q, \kappa)$ consisting of a set $Q \subseteq \Sigma^*$ of strings over $\Sigma$ and a parameterization $\kappa$ over $\Sigma^*$. A parameterized problem $\Pi$ is *fixed-parameter tractable* if there exist an algorithm $\mathcal{A}$, a computable function $f\colon \mathbf{N_0} \to \mathbf{N_0}$ and a polynomial $p$ such that for all $x \in \Sigma^*$, $\mathcal{A}$ decides $x$ in time $f(\kappa(x)) \cdot p(|x|)$.

We are, in particular, dealing with decision problems for which the input (a word $Q \subseteq \Sigma^*$ as above) is composed of a graph and an integer argument. To formally capture this form of an input, we give the next definition.

**Definition 2.2** (Graph problem)**.** A *graph problem* $\Pi$ is a set of pairs $(G, \xi)$, where $G$ is a graph and $\xi \in \mathbf{N_0}$, such that for all graphs $G_1, G_2$ and all $\xi \in \mathbf{N_0}$, $G_1 \cong G_2$ implies that $(G_1, \xi) \in \Pi$ iff $(G_2, \xi) \in \Pi$.

We remark that for graph problems which have no integer argument on the input, we may simply pad an arbitrary integer $\xi$ such as 0.

**Definition 2.3** (Kernelization)**.** A *kernelization* of a parameterized problem $(Q, \kappa)$ over the alphabet $\Sigma$ is a polynomial-time computable function $A\colon \Sigma^* \to \Sigma^*$ such that for all $x \in \Sigma^*$, we have

1. $x \in Q$ if and only if $A(x) \in Q$, and
2. $|A(x)| \leqslant g(\kappa(x))$,

where $g$ is some computable function. The function $g$ is called the *size* of the kernel. If $g(\kappa(x)) = \kappa(x)^{\mathcal{O}(1)}$ or $g(\kappa(x)) = \mathcal{O}(\kappa(x))$, we say that $\Pi$ admits a *polynomial kernel* and a *linear kernel*, respectively.

**Definition 2.4** (Treewidth)**.** A *tree decomposition* $\mathcal{T}$ of an (undirected) graph $G = (V, E)$ is a pair $(T, \chi)$, where $T$ is a tree and $\chi$ is a function that assigns each tree node $t$ a set $\chi(t) \subseteq V$ of vertices such that the following conditions hold:

(P1) For every vertex $u \in V$, there is a tree node $t$ such that $u \in \chi(t)$.

(P2) For every edge $\{u, v\} \in E(G)$ there is a tree node $t$ such that $u, v \in \chi(t)$.

(P3) For every vertex $v \in V(G)$, the set of tree nodes $t$ with $v \in \chi(t)$ forms a subtree of $T$.

The sets $\chi(t)$ are called *bags* of the decomposition $\mathcal{T}$ and, in particular, $\chi(t)$ is the bag associated with the tree node $t$. The *width* of a tree decomposition $(T, \chi)$ is the size of a largest bag minus 1. The *treewidth* of a graph $G$, denoted by $\mathbf{tw}(G)$, is the minimum width over all tree decompositions of $G$. Any tree decomposition of $G$ of width $\mathbf{tw}(G)$ is called *optimal*.

Let $\mathcal{T} = (T, \chi)$ be a tree decomposition of a graph $G$ and let $G'$ be an induced subgraph of $G$. The *projection* of $\mathcal{T}$ onto $G'$, denoted by $\mathcal{T}|G'$, is the pair $(T, \chi')$ where $\chi'(t) = \chi(t) \cap V(G')$ for every $t \in V(T)$. It is well known that $\mathcal{T}|G'$ is a tree decomposition of $G'$.

**Definition 2.5** (Pathwidth)**.** A *path decomposition* of a graph $G$ is a tree decomposition $(T, \chi)$ such that $T$ is a path. The *pathwidth* of $G$, denoted by $\mathbf{pw}(G)$, is the minimum width over all path decompositions of $G$.

All other notions and definitions introduced for tree decompositions above apply in the same way for path decompositions.

It is folklore that every bag of a path or tree decomposition is a separator in the underlying graph. We will use the following formulation of this fact.

**Proposition 2.6** (folklore)**.** *Let $\mathcal{T} = (T, \chi)$ be a tree decomposition (path decomposition) of a graph $G$, let $t \in V(T)$, and let $T_1$ and $T_2$ be two sets of nodes of $T - \{t\}$ such that $\{t\}$ separates $T_1$ from $T_2$ in $T$. Then $\chi(t)$ separates $\bigcup_{s \in T_1} \chi(s)$ from $\bigcup_{s \in T_2} \chi(s)$ in $G$.*

The definition of branchwidth is done in a slightly different manner:

**Definition 2.7** (Branchwidth)**.** A *branch-decomposition* of a graph $G$ is a pair $(T, \tau)$ where $T$ is a tree of maximum degree three and $\tau$ a bijective function $\tau : E(G) \to \{t : t \text{ is a leaf of } T\}$. For an edge $e$ of $T$, the connected components of $T \setminus e$ induce a bipartition $(X, Y)$ of the edge set of $G$. The *width* of $e$ is then defined as the number of vertices of $G$ incident both with an edge of $X$ and an edge of $Y$. The *width* of $(T, \tau)$ is the maximum width over all edges of $T$. The *branchwidth* of $G$, denoted by $\mathbf{bw}(G)$, is the minimum of the width of all branch-decompositions of $G$.

It is well know fact that the branchwidth of a graph class is bounded if and only if its treewidth is bounded.

### *2.3. Grad and graph classes of bounded expansion*

Let us recall the main definitions pertaining to the notion of graphs of bounded expansion. We follow the recent book by Nešetřil and Ossona de Mendez [6].

**Definition 2.8** (Greatest reduced average density (grad) [7, 23])**.** Let $\mathcal{G}$ be a graph class. Then the *greatest reduced average density* of $\mathcal{G}$ with *rank* $d$ is defined as

$$\nabla_d(\mathcal{G}) = \sup_{H \in \mathcal{G} \,\triangledown\, d} \frac{|E(H)|}{|V(H)|}.$$

This notation is also used for graphs via the convention that $\nabla_d(G) := \nabla_d(\{G\})$. In particular, note that $G \,\triangledown\, 0$ denotes the set of subgraphs of $G$ and hence $2\nabla_0(G)$ is the maximum average degree of all subgraphs of $G$. The *degeneracy* of $G$ is, therefore, exactly $2\nabla_0(G)$.

**Definition 2.9** (Bounded expansion [7])**.** A graph class $\mathcal{G}$ has *bounded expansion* if there exists a function $f\colon \mathbf{N_0} \to \mathbf{R}$ (called the *expansion function*) such that for all $d \in \mathbf{N_0}$, $\nabla_d(\mathcal{G}) \leqslant f(d)$.

If $\mathcal{G}$ is a graph class of bounded expansion with expansion function $f$, we say that $\mathcal{G}$ has *expansion bounded by* $f$. An important relation we make use of later is: $\nabla_d(G) = \nabla_0(G \mathbin{\triangledown} d)$, i.e. the grad of $G$ with rank $d$ is precisely one half the maximum average degree of subgraphs of its depth $d$ shallow minors.

Another important notion that we make use of extensively is that of treedepth. In this context, a *rooted forest* is a disjoint union of rooted trees. For a vertex $x$ in a tree $T$ of a rooted forest, the *height* (or *depth*) of $x$ in the forest is the number of vertices in the path from the root of $T$ to $x$. The *height of a rooted forest* is the maximum height of a vertex of the forest.

**Definition 2.10** (Treedepth)**.** Let the *closure* of a rooted forest $\mathcal{F}$ be the graph $\mathrm{clos}(\mathcal{F}) = (V_c, E_c)$ with the vertex set $V_c = \bigcup_{T\in\mathcal{F}} V(T)$ and the edge set $E_c = \{xy\colon x \text{ is an ancestor of } y \text{ in some } T \in \mathcal{F}\}$. A *treedepth decomposition* of a graph $G$ is a rooted forest $\mathcal{F}$ such that $G \subseteq \mathrm{clos}(\mathcal{F})$. The *treedepth* $\mathbf{td}(G)$ of a graph $G$ is the minimum height of any treedepth decomposition of $G$.

**Proposition 2.11** ([6])**.** *Given a graph $G$ with $n$ nodes and a constant $w$, it is possible to decide whether $G$ has treedepth at most $w$, and if so, to compute an optimal treedepth decomposition of $G$ in time $\mathcal{O}(n)$.*

We list some additional well-known facts about graphs of bounded treedepth.

**Proposition 2.12** (see, e.g. [6])**.** *Let $G$ be a graph.*

*a) If $G$ has no path with more than $d$ vertices, then $\mathbf{td}(G) \leqslant d$.*

*b) If $\mathbf{td}(G) \leqslant d$, then $G$ has no paths with $2^d$ vertices and, in particular, any DFS-tree of $G$ has depth at most $2^d - 1$.*

*c) If $\mathbf{td}(G) \leqslant d$, then $G$ is $d$-degenerate and hence has at most $d \cdot |V(G)|$ edges.*

*d) Any DFS-forest $\mathcal{F}$ of $G$ is a treedepth decomposition of $G$ (not necessarily optimal).*

*e) If $\mathcal{F}$ is a treedepth decomposition of $G$, then the vertex sets of root-to-leaf paths of each $T \in \mathcal{F}$, ordered by the DFS order of their leaf ends, form the bags of a path decomposition of $G$. Consequently, if $\mathbf{td}(G) \leqslant d$, then $\mathbf{tw}(G) \leqslant \mathbf{pw}(G) \leqslant d - 1$.*

*f) It is $\mathbf{td}(G) \leqslant d$ if and only if $G$ can be colored with at most $d$ colors such that every connected subgraph of $G$ contains at least one color that appears exactly once. Consequently, the property '$\mathbf{td}(G) \leqslant d$' is expressible in $MSO_1$ logic for each fixed value of $d$.*

A useful way of thinking about graphs of bounded treedepth is that they are (sparse) graphs with no long paths.

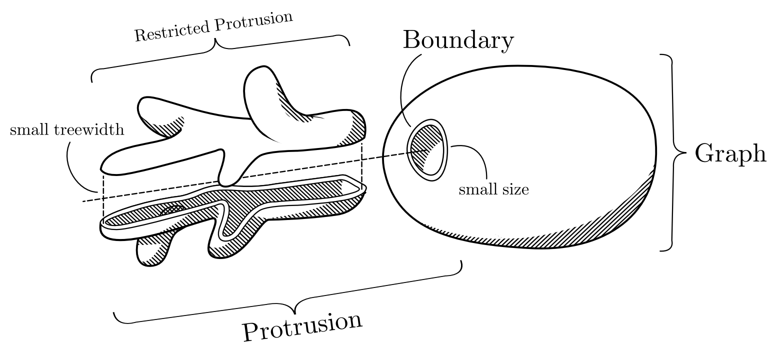


Figure 1: The anatomy of a protrusion.

For a graph $G$ and an integer $d$, a *modulator to treedepth $d$* of $G$ is a set of vertices $M \subseteq V(G)$ such that $\mathbf{td}(G - M) \leqslant d$. The size of a modulator is the cardinality of the set $M$.

Finally, we need the following well-known result on degenerate graphs.

**Proposition 2.13** ([24])**.** *Every $d$-degenerate graph $G$ with $n \geqslant d$ vertices has at most $2^d(n - d + 1)$ cliques.*

## 3. The Protrusion Machinery

In this section, we recapitulate the main ideas of the protrusion machinery developed in [1, 4].

**Definition 3.1** ($r$-protrusion [1])**.** Given a graph $G$, a set $W \subseteq V(G)$ is an *$r$-protrusion* of $G$ if $|\partial^G(W)| \leqslant r$ and $\mathbf{tw}(G[W]) \leqslant r - 1$.[3] We call $\partial^G(W)$ the *boundary* and $|W|$ the *size* of the protrusion $W$.

Thus an $r$-protrusion in a graph can be seen as an induced subgraph that is separated from the rest of the graph by a small boundary and, in addition, has small treewidth. See Figure 1.

A *$t$-boundaried graph* is a pair $(G, bd(G))$, where $G$ is a graph and $bd(G) \subseteq V(G)$ is a set of $t = |\,bd(G)|$ vertices with distinct *labels* from the set $\{1, \ldots, t\}$. The graph $G$ is called the *underlying unlabeled graph* and $bd(G)$ is called the *boundary*.[4] Given a graph class $\mathcal{G}$, we let $\mathcal{G}_t$ denote the class of $t$-boundaried graphs $(G, bd(G))$ where $G \in \mathcal{G}$.

Consider $t$-boundaried graphs $(H, bd(H))$ and $(G, bd(G))$. In the following we write $bd(G) = bd(H)$ to denote that the boundary vertices of $G$ and $H$ are the same *and* that they are labeled in the same way. We say that $(H, bd(H))$ is a subgraph of $(G, bd(G))$ if $H \subseteq G$ and $bd(H) = bd(G)$. We say that $(H, bd(H))$

[3] We want the bags in a tree-decomposition of $G[W]$ to be of size at most $r$.
[4] Usually denoted by $\partial(G)$, but this collides with our usage of $\partial$.

is an induced subgraph of $(G, bd(G))$ if for some $X \subseteq V(G)$, $H = G[X]$ and $bd(H) = bd(G)$. The boundaries of two $t$-boundaried graphs $(G, bd(G))$ and $(H, bd(H))$ are *identical* if the function mapping each vertex of $bd(G)$ to that vertex of $bd(H)$ with the same label is an isomorphism between $G[bd(G)]$ and $H[bd(H)]$. Note that in the case of $(H, bd(H))$ being an induced subgraph of $(G, bd(G))$, the boundaries are identical by definition. In the following, we will denote a $t$-boundaried graph $(G, bd(G))$ shortly by $\widetilde{G}$ to avoid cumbersome notation.

**Definition 3.2** (Gluing and ungluing)**.** For $t$-boundaried graphs $\widetilde{G}_1$ and $\widetilde{G}_2$, we let $\widetilde{G}_1 \oplus \widetilde{G}_2$ denote the graph obtained by taking the disjoint union of $G_1$ and $G_2$ and identifying each vertex in $bd(G_1)$ with the vertex in $bd(G_2)$ with the same label. The resulting order of vertices is an arbitrary extension of the orderings on $V(G_1)$ and $V(G_2) \setminus V(G_1)$. This operation is called *gluing*.

Let $H$ be an induced subgraph of $G$ and let $B$ denote the set $\partial^G(H)$ labeled with distinct labels from $\{1, \ldots, t\}$ such that $t = |B| = |\partial^G(H)|$. The operation of *ungluing* $H$ from $G$ creates the $t$-boundaried graph $G \ominus_B H := \big(G - (V(H) \setminus B), B\big)$.

The gluing operation entails taking the union of edges both of whose endpoints are in the boundary, with implicit deletion of multiple edges to keep the graph simple. The ungluing operation preserves the boundary (both the vertices and the edges). For the sake of clarity, we sometimes annotate the $\oplus$ operator with the boundary as well.

Note that an $r$-protrusion $W$ of a graph $G$ implicitly defines a $t$-boundaried graph $\widetilde{G}[W] := \big(G[W], \partial^G(W)\big)$, $t = |\partial^G(W)| \leqslant r$, where the boundary vertices are assigned labels from $\{1, \ldots, t\}$ according to their order in $G$. Hence we can rigorously deal with protrusions in $G$ as with $t$-boundaried subgraphs of $G$ as, e.g., in the following definition.

**Definition 3.3** (Replacement)**.** Let $W$ be an $r$-protrusion of a graph $G$ defining the $t$-boundaried graph $\widetilde{G}[W]$, and let $B$ be the labeled set of the boundary $\partial^G(W)$. For a $t$-boundaried graph $\widetilde{H}$, *replacing* $\widetilde{G}[W]$ by $\widetilde{H}$ in $G$ is defined as the operation $(G \ominus_B G[W]) \oplus_B \widetilde{H}$.

The following definition concerns the centerpiece of our framework. Recall that an equivalence relation has *finite index* if it defines a finite number of equivalence classes.

**Definition 3.4** (Finite integer index; FII)**.** Let $\Pi$ be a graph problem and let $\widetilde{G}_1 = (G_1, bd(G_1))$, $\widetilde{G}_2 = (G_2, bd(G_2))$ be two $t$-boundaried graphs. We say that $\widetilde{G}_1 \equiv_{\Pi,t} \widetilde{G}_2$ if there exists an integer constant $\Delta_{\Pi,t}(\widetilde{G}_1, \widetilde{G}_2)$ such that for all $t$-boundaried graphs $\widetilde{H} = (H, bd(H))$ and for all $\xi \in \mathbf{N_0}$:

$$\big(\widetilde{G}_1 \oplus \widetilde{H}, \xi\big) \in \Pi \text{ iff } \big(\widetilde{G}_2 \oplus \widetilde{H}, \xi + \Delta_{\Pi,t}(\widetilde{G}_1, \widetilde{G}_2)\big) \in \Pi.$$

We say that $\Pi$ has *finite integer index in the class* $\mathcal{F}$ if, for every $t \in \mathbf{N_0}$, the relation $\equiv_{\Pi,t}$ has finite index if restricted to $\mathcal{F}$.

Note that the constant $\Delta_{\Pi,t}(\widetilde{G}_1, \widetilde{G}_2)$ depends on $\Pi$, $t$, and the *ordered pair* $(\widetilde{G}_1, \widetilde{G}_2)$ so that $\Delta_{\Pi,t}(\widetilde{G}_1, \widetilde{G}_2) = -\Delta_{\Pi,t}(\widetilde{G}_2, \widetilde{G}_1)$. On most occasions, the problem $\Pi$ and the class $\mathcal{F}$ will be clear from the context and in such situations, we use $\equiv_t$ and $\Delta_t$ instead of $\equiv_{\Pi,t}$ and $\Delta_{\Pi,t}$, respectively.

If a graph problem has finite integer index then its instances can be reduced by "replacing protrusions". The technique of replacing protrusions hinges on the fact that each protrusion of "large" size can be replaced by a "small" gadget from the same equivalence class as the protrusion, which consequently behaves similarly w.r.t. the problem at hand. If $\widetilde{G}_1$ is replaced by a gadget $\widetilde{G}_2$ (strictly saying, $H \oplus \widetilde{G}_1$ is replaced by $H \oplus \widetilde{G}_2$), then $\xi$ changes by $\Delta_{\Pi,t}(\widetilde{G}_1, \widetilde{G}_2)$. Many problems have finite integer index in general graphs, including Vertex Cover, Independent Set, Feedback Vertex Set, Dominating Set, Connected Dominating Set, and Edge Dominating Set. For a more complete list see [1, 4]. Some problems that do not have finite integer index in general graphs are Connected Feedback Vertex Set, Longest Path and Longest Cycle.

For a graph class $\mathcal{F}$, let $\mathcal{F}_t$ denote the class of all $t$-boundaried graphs made of the members of $\mathcal{F}$. The next lemma shows that if we assume that a graph problem $\Pi$ has FII in a graph class $\mathcal{F}$, then we can choose finitely many representatives for the equivalence classes of $\equiv_{\Pi,t}$ from a (possibly different) graph class $\mathcal{G}$ under certain circumstances.

**Lemma 3.5.** *Let $\mathcal{F}$ be a graph class and $\Pi$ a graph problem such that $\Pi$ has FII in $\mathcal{F}$. Let $\mathcal{G}$ be a class of graphs in which some vertices have labels from $\{1, \ldots, t\}$, and $\preceq$ be a relation on $\mathcal{G}$ such that $\mathcal{G}$ is well-quasi-ordered by $\preceq$. Then, for each $t \in \mathbf{N_0}$, there exists a finite set $\mathcal{R}(t, \mathcal{F}, \mathcal{G}, \preceq) \subseteq \mathcal{F}_t \cap \mathcal{G}$ with the following property. For every $\widetilde{G} = (G, bd(G)) \in \mathcal{F}_t \cap \mathcal{G}$ there exists $\widetilde{G}_0 = (G_0, bd(G_0)) \in \mathcal{R}(t, \mathcal{F}, \mathcal{G}, \preceq)$ such that $bd(G)$ and $bd(G_0)$ are identical, $\widetilde{G} \equiv_{\Pi,t} \widetilde{G}_0$, and $G_0 \preceq G$.*

*Proof.* Let $\mathcal{E}_1, \ldots, \mathcal{E}_q$ be the equivalence classes of the relation $\equiv_{\Pi,t}$ on $\mathcal{F}_t$, where $q$ is some constant. For each equivalence class $\mathcal{E}_i$, define $\mathcal{E}'_i = \mathcal{E}_i \cap \mathcal{G}$. Next, partition $\mathcal{E}'_i$ into at most $2^{t^2} \cdot t!$ sets $\mathcal{E}'_{i,j}$ such that all graphs in $\mathcal{E}'_{i,j}$ have identical boundaries. Since $\mathcal{G}$ is well-quasi-ordered by $\preceq$, there is a *finite* set $\mathcal{G}_{i,j} \subseteq \mathcal{E}'_{i,j}$ of the $\preceq$-minimal elements, for every $i, j$ as above. In other words, for all $\widetilde{G} \in \mathcal{E}'_{i,j}$ there exist $\widetilde{G}_0 \in \mathcal{G}_{i,j}$ satisfying the three properties stated in the lemma. Consequently, $\bigcup_j \mathcal{G}_{i,j}$ can be chosen as the representatives for each $\mathcal{E}_i$. Altogether, define $\mathcal{R}(t, \mathcal{F}, \mathcal{G}, \preceq) = \bigcup_{i,j} \mathcal{G}_{i,j}$. Since $\mathcal{R}(t, \mathcal{F}, \mathcal{G}, \preceq)$ is the finite union of finite sets, it is finite. $\square$

Let us explain how we use Lemma 3.5. The graph problems $\Pi$ that we consider in this paper usually have FII on the class of general graphs or, for all $p \in \mathbf{N_0}$, in the class of graphs of treedepth at most $p$. In accordance with the notation in Lemma 3.5, the class $\mathcal{F}$ corresponds to the class where $\Pi$ has FII. The choice of our parameter now ensures that our kernelization replaces protrusions of treedepth at most a previously fixed constant $d$: choosing $\mathcal{G}$ to be the graphs of treedepth at most $d$, all protrusions (actually the graphs induced by them) are members of $\mathcal{F} \cap \mathcal{G}$. As $\mathcal{G}$ is well-quasi ordered under the label-preserving induced subgraph relation [6, Chapter 6, Lemma 6.13], we choose $\preceq$ to be $\subseteq_{\text{ind}}$.

Now consider a restriction of the graph problem $\Pi$ to a class $\mathcal{K}$ that is closed under taking induced subgraphs. In this paper, the class $\mathcal{K}$ is a hereditary graph class of bounded expansion or a hereditary and nowhere dense class. This ensures that $\emptyset \neq \mathcal{K} \cap \mathcal{G} \subseteq \mathcal{F} \cap \mathcal{G}$. Given an instance $(G, \xi)$ of $\Pi$ with $G \in \mathcal{K}$, one can replace a protrusion of $G$ by a representative (of constant size) that is an *induced subgraph* of that protrusion, ensuring that this replacement creates a graph that still resides in $\mathcal{K}$. To summarize, Lemma 3.5 guarantees that the protrusion replacement rule (described next) preserves the graph class $\mathcal{K}$ and the parameter.

As preparation for the kernelization theorems of the next section, let $\mathfrak{P}$ denote the set of all graph problems that have FII on general graphs or, for each $p \in \mathbf{N_0}$, in the class of graphs of treedepth at most $p$. Our reduction rule is formalized as follows.

**Reduction Rule 3.6** (Protrusion replacement)**.** Let $t, d \in \mathbf{N_0}$ and let $\Pi \in \mathfrak{P}$. Let $\mathcal{R}(t, d)$ be a class of boundaried graphs of treedepth at most $d$ containing representatives of the equivalence classes of $\equiv_{\Pi,i}$ restricted to the graphs of treedepth at most $d$, for $i = 1, \ldots, t$. Let $(G, \xi)$ be an instance of $\Pi$ and assume that $W \subseteq V(G)$ is a $t$-protrusion of treedepth at most $d$ with boundary $B = \partial^G(W)$ in $G$ of size $i = |B| \leqslant t$. Let $\widetilde{R} \in \mathcal{R}(t, d)$ be a $\equiv_{\Pi,i}$-representative of $\widetilde{G}[W]$. The protrusion replacement rule is the following:

$$\text{Reduce } (G, \xi) \text{ to } \ (G', \xi') := \big((G \ominus_B G[W]) \oplus_B \widetilde{R},\ \xi + \Delta_{\Pi,i}(\widetilde{G}[W], \widetilde{R})\big).$$

We let $\mathcal{F}$ denote the class on which the problem has FII and by $\mathcal{G}$ the class of graphs of treedepth at most $d$. The existence of a suitable *finite set of representatives* $\mathcal{R}(t, d)$ for Rule 3.6 is guaranteed by Lemma 3.5: we let $\mathcal{R}(t, d)$ denote the finite set $\bigcup_{i=1}^{t} \mathcal{R}(i, \mathcal{F}, \mathcal{G}, \subseteq_{\text{ind}})$ from Lemma 3.5, and $\rho(t, d)$ denote the size of the largest member of $\mathcal{R}(t, d)$. The safety of the protrusion replacement follows from the definition of FII.

In what follows, when applying the protrusion replacement by Rule 3.6, we will always assume that for the fixed problem $\Pi \in \mathfrak{P}$ and each $t, d \in \mathbf{N_0}$, we are given the finite set $\mathcal{R}(t, d)$ of representatives. Our algorithm is therefore non-constructive, as are all previous algorithms in the meta-kernelization line of work [1, 4, 25, 26].

## 4. Linear Kernels on Graphs of Bounded Expansion

In this section we provide the underlying meta-theorems of our new kernelization results. Namely, we are going to show that graph problems that have finite integer index on general graphs or in the class of graphs with bounded treedepth admit linear kernels on hereditary graph classes with bounded expansion, when parameterized by the size of a modulator to constant treedepth. On nowhere dense classes, we obtain almost-linear kernels.

Our main theorem is the following.

**Theorem 4.1.** *Let $\mathcal{K}$ be a graph class of bounded expansion and let $d \in \mathbf{N_0}$ be a constant. Let $\Pi \in \mathfrak{P}$. Then there is an algorithm that takes as input $(G, \xi) \in \mathcal{K} \times \mathbf{N_0}$ and, in time $\mathcal{O}(|G| + \log \xi)$, outputs $(G', \xi')$ such that*

1. *$(G, \xi) \in \Pi$ if and only if $(G', \xi') \in \Pi$;*
2. *$G'$ is an induced subgraph of $G$; and*
3. *$|G'| = \mathcal{O}(|S|)$, where $S$ is an optimal treedepth-$d$ modulator of the graph $G$.*

We proceed as follows. Because an optimal treedepth-$d$ modulator cannot be assumed as part of the input, we compute an approximate modulator $S \subseteq V(G)$ to partition $V(G)$ into sets $Y_0 \uplus Y_1 \uplus \cdots \uplus Y_\ell$ such that $S \subseteq Y_0$ and $|Y_0| = \mathcal{O}(|S|)$ and for $1 \leqslant i \leqslant l$, $Y_i$ induces a collection of connected components of $G - Y_0$ that have exactly the same *small* neighborhood in $Y_0$. We then use bounded expansion to show that $\ell = \mathcal{O}(|S|)$ and use protrusion reduction to replace each $G[Y_i]$, $1 \leqslant i \leqslant l$, by an *induced subgraph* of $G[Y_i]$ of constant size. Every time the protrusion replacement rule is applied, $\xi$ is modified. This results in an equivalent instance $(G', \xi')$ such that $G' \subseteq G$ and $|G'| = \mathcal{O}(|S|)$, as claimed in Theorem 4.1.

**Lemma 4.2.** *Fix $d \in \mathbf{N_0}$. Given a graph $G$, one can in $O(|G|^2)$ time compute a subset $S \subseteq V(G)$ such that $\mathbf{td}(G - S) \leqslant d$ and $|S|$ is at most $2^d$ times the size of an optimal treedepth-$d$ modulator of $G$. On graph classes of bounded expansion, the set $S$ can be computed in linear time. On nowhere dense classes it can be computed in time $O(|G|^{1+\varepsilon})$ for every fixed $\varepsilon > 0$.*

*Proof.* We use the fact, cf. Proposition 2.12 b), that any DFS-tree of a graph of treedepth $d$ has height at most $2^d - 1$. Start with $S_0 = \emptyset$. Compute a DFS-tree of the graph $G$ and if it has height more than $2^d - 1$, then $\mathbf{td}(G) > d$. So, we take some path $P$ from the root of the tree of length $2^d - 1$ and add all the $2^d$ vertices of $P$ into a set $S_0$ (to be part of $S$), delete $V(P)$ from the graph and repeat. (Clearly, at least one of the vertices of $P$ must be in any modulator.) At the end of this procedure, the DFS-tree of the remaining graph $G - S_0$ has height at most $2^d - 1$. By Proposition 2.12 e), this gives a path decomposition of the graph of width at most $2^d - 2$. Now use standard tools (e.g., Courcelle's theorem [27] via Proposition 2.12 f) to obtain an optimal treedepth-$d$ modulator $S_1$ in $G - S_0$, and set $S = S_0 \cup S_1$. Since the treewidth of $G - S_0$ is a constant, the latter algorithm runs in time linear in the size of the graph. The overall size of the modulator is at most $2^d$ times the optimal solution.

For a graph $G$ from a class of bounded expansion, we modify the way $S_0$ is computed above (the resulting set will not be larger than the one computed above, and often much smaller). By [28], graph classes of bounded expansion admit low treedepth coloring: Given any integer $p$, there exists an integer $n_p$ such that any graph of the class can be properly vertex colored using $n_p$ colors such that for any set of $1 \leqslant i \leqslant p$ colors, the graph induced by the vertices that receive these $i$ colors has treedepth at most $i$. Such a coloring is called a $p$-treedepth coloring and can be computed in linear time [28]. Here we choose $p = 2^d$ and obtain such a coloring for $G$ using $n_p$ colors. Let $G_1, \ldots, G_r$ denote the subgraphs induced by at most $2^d$ of these color classes where $r < 2^{n_p} = \mathcal{O}(1)$. Note that $\sum_j |G_j| = \mathcal{O}(|G|)$, since every vertex of $G$ appears in at most a constant number of subgraphs and since $G$ has constant degeneracy, $\sum_j |E(G_j)| = \mathcal{O}(|G|)$ as well.

Any path in $G$ of length $2^d - 1$ must be in some subgraph $G_j$, for $1 \leqslant j \leqslant r$, and we hit all such paths with a set $S_0$ obtained in the following iterated procedure.

Start with $S_0 = \emptyset$. For each $j = 1, 2, \ldots, r$, we simply construct a treedepth decomposition of $G_j - S_0$, e.g., by depth-first search. Using standard dynamic programming we find an optimum hitting set for the set of all paths of length $2^d - 1$ in $G_j - S_0$ and add its vertices into $S_0$ (and delete them from the graph). Again, some hitting set for these paths must be in any modulator. The time taken to do this for each subgraph $G_j - S_0$ is $\mathcal{O}(|E(G_j)|)$. The total time taken is therefore $\sum_j |E(G_j)| = \mathcal{O}(|G|)$.

The approach for nowhere dense classes is nearly the same: by [7, 6], for a nowhere dense class $\mathcal{G}$ and $\varepsilon' > 0$, $p \in \mathbf{N_0}$ there exists a threshold $N_{\varepsilon',p}$ such that for all $G \in \mathcal{G}$ with $|G| \geqslant N_{\varepsilon',p}$ it holds that $G$ has a $p$-treedepth coloring with at most $|G|^{\varepsilon'}$ colors. By a similar statement, nowhere dense graphs are $|G|^{1+\varepsilon'}$-degenerate. Therefore, for every $\varepsilon > 0$, the above algorithm runs in time $O(|G|^{1+\varepsilon})$ by choosing $\varepsilon' = \varepsilon/p$ and $p = 2^d$; now the subgraphs $G_1, \ldots, G_r$ induced by at most $2^d$ colors have again treedepth at most $2^d$ while $r \leqslant (|G|^{\varepsilon'})^p = |G|^{\varepsilon}$. Using the fact that nowhere dense graphs have degeneracy $|G|^{1+\varepsilon}$, we conclude that the running time to construct $S_0$ is $\sum_j |E(G_j)| = \mathcal{O}(|G|^{1+\varepsilon})$ and this also bounds the total running time. $\square$

We will make heavy use of the following lemma to prove the kernel size.

**Lemma 4.3.** *Let $G = (X, Y, E)$ be a bipartite graph, and $p \geqslant \nabla_1(G)$. Then there are at most*

1. *$2p \cdot |X|$ vertices in $Y$ with degree greater than $2p$;*
2. *$(4^p + 2p) \cdot |X|$ distinct subsets $X' \subseteq X$ such that $X' = N(u)$ for some $u \in Y$.*

*Proof.* We construct a sequence of graphs $G_0, G_1, \ldots, G_\ell$ such that $G_i \in G \mathbin{\triangledown} 1$ for all $0 \leqslant i \leqslant \ell$ as follows. Set $G_0 = G$, and for $0 \leqslant i \leqslant \ell - 1$ construct $G_{i+1}$ from $G_i$ by choosing a vertex $v \in V(G_i) \setminus X$ such that $N(v) \subseteq X$ contains two non-adjacent vertices $u, w$ in $G_i$ and contract $uv$ into the vertex $u$ to obtain $G_{i+1}$. Recall that contracting $uv$ to $u$ is equivalent to deleting vertex $v$ and adding edges between each vertex in $N(v) \setminus u$ and $u$. Note that this contraction will only add edges to $X$ and remove vertices from $Y$. Hence, for $0 \leqslant i \leqslant \ell$, we maintain $X \subseteq V(G_i) \subseteq X \cup Y$ and that $V(G_i) \setminus X \subseteq Y$ is an independent set.

This process clearly terminates, as $G_{i+1}$ has at least one more edge between vertices of $X$ than $G_i$. Note that $G_i \in G \triangledown 1$ for $0 \leqslant i \leqslant \ell$, as the edges $e_1, \ldots, e_{i-1}$ that were contracted to vertices in $X$ in order to construct $G_i$ had one endpoint each in $X$ and $Y$, the endpoint in $Y$ being deleted after each contraction. Thus, $e_1, \ldots, e_{i-1}$ induce a set of stars in $V(G) = V(G_0)$, and $G_i$ is obtained from $G$ by contracting these stars. We therefore conclude that $G_i$ is a depth-one shallow minor of $G$. In particular, this implies $G_\ell[X]$ is $2p$-degenerate and has at most $2p|X|$ edges. Further, note that for each $0 \leqslant i \leqslant \ell$, $Y \cap V(G_i)$ is, by construction, still an independent set in $G_i$.

Let us now prove the first claim. To this end, assume that there is a vertex $v \in Y \cap V(G_\ell)$ such that $\deg(v) > 2p$. We claim that $G_\ell[N(v)]$ (where $N(v) \subseteq X$) is a clique. If not, we could choose a pair of non-adjacent vertices in $G_\ell[N(v)]$ and construct a $(\ell+1)$-th graph for the sequence which would contradict the fact that $G_\ell$ is the last graph of the sequence. However, a clique of size $|\{v\} \cup N(v)| > 2p+1$ is not $2p$-degenerate. Hence we conclude that no vertex of $Y \cap V(G_\ell)$ has degree larger than $2p$ in $G_\ell$ (and in $G$). Therefore the vertices of $Y$ of degree greater than $2p$ in the graph $G$, if there were any, must have been deleted during the edge contractions that resulted in the graph $G_\ell$. As every contraction added at least one edge between vertices in $X$ and since $G_\ell[X]$ contains at most $2p|X|$ edges, the first claim follows.

For the second claim, consider the set $Y' = Y \cap V(G_\ell)$. The neighborhood of every vertex $v \in Y'$ induces a clique in $G_\ell[X]$. From the $2p$-degeneracy of $G_\ell[X]$ and Proposition 2.13, it follows that $G_\ell[X]$ has at most $2^{2p}|G_\ell[X]| = 4^p \cdot |X|$ cliques. Thus the number of subsets of $X$ that are neighborhoods of vertices in $Y$ in $G$ is at most $(4^p + 2p) \cdot |X|$, where we accounted for vertices of $Y$ lost via contractions by the bound on the number of edges in $G_\ell[X]$. $\square$

The following two corollaries to Lemma 4.3 show how it can be applied in our situation.

**Corollary 4.4.** *Let $\mathcal{K}$ be a graph class whose expansion is bounded by a function $f\colon \mathbf{N_0} \to \mathbf{R}$. Suppose that for $G \in \mathcal{K}$ and $S \subseteq V(G)$, $C_1, \ldots, C_s$ are disjoint connected subgraphs of $G - S$ satisfying the following two conditions for $1 \leqslant i \leqslant s$:*

1. *$diam(C_i) \leqslant \delta$ and*
2. *$|N_S(C_i)| > 2 \cdot f(\delta + 1)$.*

*Then $s \leqslant 2 \cdot f(\delta + 1) \cdot |S|$.*

*Proof.* We construct an auxiliary bipartite graph $\bar{G}$ with partite sets $S$ and $Y = \{C_1, \ldots, C_s\}$. There is an edge between $C_i$ and $x \in S$ iff $x \in N_S(C_i)$. Note that $\bar{G}$ is a depth-$\delta$ shallow minor of $G$ with branch sets $C_i, 1 \leqslant i \leqslant s$. In relation to Lemma 4.3 we would like to show that, for any $F \in \bar{G} \triangledown 1$, it is $F \in G \triangledown (\delta + 1)$ (while $\triangledown$ is not additive in general). This follows since a branch set of $F$ in $G$ is induced by a vertex of $S$ plus a subcollection of attached sets

$C_i$, $1 \leqslant i \leqslant s$, or by one set $C_i$ and a subset of attached vertices from $S$. In both the cases the radius is at most $1 + \max_i diam(C_i) \leqslant \delta + 1$.

Consequently, $\nabla_1(\bar{G}) \leqslant \nabla_{\delta+1}(G) \leqslant f(\delta + 1)$ and, by Lemma 4.3 for the choice $p = f(\delta + 1)$,

$$s \leqslant 2p|S| = 2f(\delta + 1) \cdot |S|.$$

□

**Corollary 4.5.** *Let $\mathcal{K}$ be a graph class whose expansion is bounded by a function $f\colon \mathbf{N_0} \to \mathbf{R}$. Suppose that for $G \in \mathcal{K}$ and $S \subseteq V(G)$, $\mathcal{C}_1, \dots, \mathcal{C}_t$ are sets of connected components of $G - S$ such that for all pairs $C, C' \in \bigcup_i \mathcal{C}_i$ it holds that $C, C' \in \mathcal{C}_j$ for some $j$ if and only if $N_S(C) = N_S(C')$. Let $\delta > 0$ be a bound on the diameter of the components, i.e. for all $C \in \bigcup_i \mathcal{C}_i$, $diam(C) \leqslant \delta$. Then there can be only at most $t \leqslant (4^{f(\delta+1)} + 2f(\delta + 1)) \cdot |S|$ such sets $\mathcal{C}_i$.*

*Proof.* As in the proof of Corollary 4.4, we construct a bipartite graph $\bar{G}$ with partite sets $S$ and $Y = \{C_1, \dots, C_r\}$, and argue about $\nabla_1(\bar{G}) \leqslant \nabla_{\delta+1}(G) \leqslant f(\delta + 1)$. By Lemma 4.3, for $p = f(\delta + 1)$,

$$t \leqslant (4^p + 2p)|S| = (4^{f(\delta+1)} + 2f(\delta + 1)) \cdot |S|.$$

□

In the first phase, our kernelization algorithm partitions an input graph according to a low-treedepth modulator (as found in Lemma 4.2).

**Lemma 4.6.** *Let $\mathcal{K}$ be a graph class with expansion bounded by $f$, $G \in \mathcal{K}$ and $S \subseteq V(G)$ be a treedepth-$d$ modulator ($d$ a constant). There is an algorithm that partitions $V(G)$ in time $\mathcal{O}(|G|)$ into sets $Y_0 \uplus Y_1 \uplus \cdots \uplus Y_\ell$ such that the following hold:*

1. *$S \subseteq Y_0$ and $|Y_0| = \mathcal{O}(|S|)$;*
2. *for $1 \leqslant i \leqslant \ell$, $Y_i$ induces a set of connected components of $G - Y_0$ that have the same neighborhood in $Y_0$ of size at most $2^{d+1} + 2 \cdot f(2^d)$;*
3. *$\ell \leqslant \left(4^{f(2^d)} + 2f(2^d)\right) \cdot |S| = \mathcal{O}(|S|)$.*

*Proof.* We first construct a DFS-forest $\mathcal{D}$ of $G - S$. Since $\mathbf{td}(G - S) \leqslant d$, the height of every tree in $\mathcal{D}$ is at most $2^d - 1$. Assume that there are $q$ trees $T_1, \dots, T_q$ in this forest rooted at $r_1, \dots, r_q$, respectively. We construct, following the same idea as in Proposition 2.12 e), for each $T_i$, $1 \leqslant i \leqslant q$, a path decomposition of the subgraph of $G[V(T_i)]$. Suppose that $T_i$ has leaves $l_1, \dots, l_s$ ordered according to their DFS-number. For $1 \leqslant j \leqslant s$, create a bag $B_j$ containing the vertices on the unique path from $l_j$ to $r_i$ and string these bags together in the order $B_1, \dots, B_s$. This is a path decomposition $\mathcal{P}_i$ of $G[V(T_i)]$ with width at most $2^d - 2$. Note that the root $r_i$ is in every bag of $\mathcal{P}_i$. These steps are in a simplified way depicted in the first loop of Algorithm 2 and clearly run in linear time.

We now use a marking algorithm similar to the one in [5] to mark $\mathcal{O}(|S|)$ bags in the path decompositions $\mathcal{P}_1, \dots, \mathcal{P}_q$ with the property that each marked

**Algorithm 2:** BAG MARKING ALGORITHM

**Input**: A graph $G$, a subset $S \subseteq V(G)$ such that $\mathbf{td}(G-S) \leqslant d$, and an integer $t > 0$.

Set $\mathcal{M} \leftarrow \emptyset$ as the set of marked bags;
**for** *each connected component* $C$ *of* $G-S$ *such that* $N_S(C) \geqslant t$ **do**
  Choose an arbitrary vertex $v \in V(C)$ as a root and construct a DFS-tree starting at $v$;
  Use the DFS-tree to obtain a path-decomposition $\mathcal{P}_C = (P_C, \mathcal{B}_C)$ of width at most $2^d - 2$ in which the bags are ordered from left to right;

Repeat the following loop for the path-decomposition $\mathcal{P}_C$ of every $C$;
**while** $\mathcal{P}_C$ *contains an unprocessed bag* **do**
  Let $B$ be the leftmost unprocessed bag of $\mathcal{P}_C$;
  Let $G_B$ denote the subgraph of $G$ induced by the vertices in the bag $B$ and in all bags to the left of it in $\mathcal{P}_C$.
  **[Large-subgraph marking step]**
  **if** $G_B$ *contains a connected component* $C_B$ *such that* $|N_S(C_B)| \geqslant t$ **then**
    $\mathcal{M} \leftarrow \mathcal{M} \cup \{B\}$ and remove the vertices of $B$ from every bag of $\mathcal{P}_C$;
  Bag $B$ is now processed;

**return** $Y_0 = S \cup V(\mathcal{M})$;

bag can be uniquely identified with a connected subgraph of $G-S$ that has a large neighborhood in the modulator $S$.

Let us set $t := 2 \cdot f(2^d) + 1$ as the threshold for what we consider a *large neighborhood* in the set $S$ and run Algorithm 2 with this value of $t$. Note that there is a one-to-one correspondence between marked bags $\mathcal{M}$ and a maximal set of pairwise disjoint, connected subgraphs with a neighborhood of size at least $t$ in $S$. Moreover each such connected subgraph has treedepth at most $d$ and hence diameter at most $2^d - 1$. By Corollary 4.4, the number of connected subgraphs of large neighborhood and hence the number of marked bags is at most $2 \cdot f(2^d - 1 + 1) \cdot |S| = \mathcal{O}(|S|)$. We set $Y_0 := V(\mathcal{M}) \cup S$. As the marking stage of the algorithm runs through every path-decomposition of the components of $G-S$ exactly once, this phase takes only linear time.

Now observe that each connected component in $G - Y_0$ has less than $t = 2 \cdot f(2^d) + 1$ neighbors in $S$: for every connected subgraph $C$ with at least $t$ neighbors in $S$, there exists a marked bag $B$ that contains at least one vertex of $C$. Importantly, the bag $B$ was the *first* bag that was marked before the number of neighbors in $S$ of *any* connected subgraph reached the threshold $t$. Hence each connected component of $G[V(C) \setminus B]$ has degree less than $t$ in $S$. Since every component is connected to at most two marked bags (in $Y_0$) and since each bag is of size at most $2^d - 1$, the size of the neighborhood of every component of $G - Y_0$ in $Y_0$ is at most $2(2^d - 1) + t \leqslant 2^{d+1} + 2 \cdot f(2^d)$.

To complete the proof, we simply cluster the connected components of $G - Y_0$

according to their neighborhoods in $Y_0$ to obtain the sets $Y_1, \dots, Y_\ell$. Since each connected component of $G - S$ is of diameter $\delta \leqslant 2^d - 1$, by Corollary 4.5, the number $\ell$ of clusters is at most $\left(4^{f(2^d)} + 2f(2^d)\right) \cdot |S| = \mathcal{O}(|S|)$, as claimed.

To accomplish this feat in linear time, we assume an arbitrary order of the vertices in $Y_0$ (say, the order in which they appear in the encoding of the graph). A simple bipartite auxiliary graph with partitions $Y_0$ and $V(G) \setminus Y_0$ with edge set $(Y_0 \times V(G) \setminus Y_0) \cap E(G)$ can be used to find the neighbors of a vertex $v \notin Y_0$ inside $Y_0$ in constant time as the number of neighbors of such a vertex is at most $2^{d+1} + 2 \cdot f(2^d) = \mathcal{O}(1)$. Thus, computing the neighbors in $Y_0$ of every component of $G - Y_0$ takes only linear time. If we store the neighborhoods of these components as lists sorted according to the ordering of $Y_0$ inside an array of length $O(|S|)$, we can sort this array in linear time using bucket sort: each entry of the list has encoding length at most $\log |V(G)|$, therefore they can be compared in constant time under the usual RAM-model. The clusters can then be simply read from the array. Thus the clustering of the components of $G - Y_0$ and therefore the whole decomposition is linear-time computable. $\square$

To prove a linear kernel, all that is left to show is that each cluster $Y_i$, $1 \leqslant i \leqslant \ell$, can be reduced to constant size. Note that each cluster is separated from the rest of the graph via a small set of vertices in $Y_0$ and that each component of $G - Y_0$ has constant treedepth even when its boundary is included. These facts enable us to use the protrusion reduction rule.

In the proof of the following lemma it will be convenient to use the following normal form of tree decompositions: A triple $(T, \{W_x \mid x \in V(T)\}, r)$ is a *nice tree decomposition* of a graph $G$ if $(T, \{W_x \mid x \in V(T)\})$ is a tree decomposition of $G$, the tree $T$ is rooted at node $r \in V(T)$, and each node of $T$ is of one of the following four types:

1. a *leaf node*: a node having no children and containing exactly one vertex in its bag;
2. a *join node*: a node $x$ having exactly two children $y_1, y_2$, and $W_x = W_{y_1} = W_{y_2}$;
3. an *introduce node*: a node $x$ having exactly one child $y$, and $W_x = W_y \cup \{v\}$ for a vertex $v$ of $G$ with $v \notin W_y$; or
4. a *forget node*: a node $x$ having exactly one child $y$, and $W_x = W_y \setminus \{v\}$ for a vertex $v$ of $G$ with $v \in W_y$.

Given a tree decomposition of a graph $G$ of width $w$, one can effectively obtain in time $\mathcal{O}(|V(G)|)$ a nice tree decomposition of $G$ with $\mathcal{O}(|V(G)|)$ nodes and of width at most $w$ [29].

For the next statement and proofs, recall the following concepts from (the end of) Section 3: the problem class $\mathfrak{P}$ and our fixed problem $\Pi \in \mathfrak{P}$, the (implicitly given) finite sets $\mathcal{R}(t, d)$ of representatives of the equivalence classes of the relations $\equiv_{\Pi,i}$, $i = 1, \dots, t$, restricted to graphs of treedepth $\leqslant d$, and $\rho(t, d)$ the size of the largest member(s) of $\mathcal{R}(t, d)$.

**Lemma 4.7.** *For fixed $d, h \in \mathbf{N_0}$ (constants) and $\mathcal{K}$ a graph class, let $(G, \xi)$ be an instance of $\Pi$ with $G \in \mathcal{K}$ and let $S \subseteq V(G)$ be a treedepth-$d$ modulator of $G$. Let $Y_0 \uplus Y_1 \uplus \cdots \uplus Y_\ell$ be a vertex partition of $V(G)$ such that*

- $S \subseteq Y_0$,
- $N(Y_i) \subseteq Y_0$ *and* $|N_{Y_0}(Y_i)| \leqslant h$ *for* $1 \leqslant i \leqslant \ell$, *and*
- $N_{Y_0}(Y_i) \neq N_{Y_0}(Y_j)$ *for* $i \neq j$.

*Then one can in $\mathcal{O}(|G| + \log \xi)$ time obtain an instance $(G', \xi')$ and a vertex partition $Y_0' \uplus Y_1' \uplus \cdots \uplus Y_\ell'$ of $V(G')$ such that*

1. $(G, \xi) \in \Pi$ *if and only if* $(G', \xi') \in \Pi$;
2. $G'$ *is an induced subgraph of* $G$ *with* $Y_0' = Y_0$; *and*
3. *for* $1 \leqslant i \leqslant \ell$ *it is* $|N_{Y_0'}(Y_i')| \leqslant h$, $\mathbf{td}(G[Y_i']) \leqslant d$, *and* $|Y_i'| \leqslant \rho(d + h, d + h) = \mathcal{O}(1)$.

*Proof.* Since $S \subseteq Y_0$ is a treedepth-$d$ modulator, for all $1 \leqslant i \leqslant \ell$, we have $\mathbf{td}(G[Y_i]) \leqslant d$ and hence $\mathbf{tw}(G[Y_i]) \leqslant d - 1$. Moreover treedepth at most $d$ implies diameter at most $2^d - 1$ for each component. Since $Y_0' = Y_0$, let $N(X)$ stand for $N_{Y_0}(X) = N_{Y_0'}(X)$. For each index $1 \leqslant i \leqslant \ell$, our algorithm constructs a tree-decomposition of $G[Y_i \cup N(Y_i)]$ of width $d - 1 + h$ that satisfies certain additional properties that we mention below.

The algorithm then uses the tree-decomposition to replace $Y_i$ in a systematic manner using the protrusion replacement Rule 3.6. The special properties of the tree-decomposition enable the algorithm to perform this replacement in $\mathcal{O}(|Y_i \cup N(Y_i)|)$ time. Total time used to replace all sets $Y_i$ is $\sum_{i=1}^{\ell} |Y_i \cup N(Y_i)|$. Since, by Corollary 4.5 (with $Y_0$ in the place of $S$), $\sum_{i=1}^{\ell} |N(Y_i)| = \mathcal{O}(\ell) = \mathcal{O}(|Y_0|)$, the running time is indeed $\mathcal{O}(|G|)$. It therefore suffices to specify the properties of our tree-decompositions and describe how each $Y_i$ is replaced with $Y_i'$.

The desired tree-decomposition $\mathcal{T}_i = \big(T_i, \{W_x \mid x \in V(T_i)\}\big)$ of width at most $d + h - 1$ for $G[Y_i \cup N(Y_i)]$ satisfies the following conditions:

1. there is a node $r \in V(T_i)$ such that $W_r = N(Y_i)$; and
2. the tree-decomposition is nice and rooted at $r$.

We use Bodlaender's linear-time algorithm [30] to compute the tree decomposition. To ensure the first condition, we simply modify the graph $G_i$ so that $N(Y_i)$ induces a clique, and then introducing an extra node $r$ in the resulting tree decomposition if no such node exists already. The niceness of the decomposition can simply be restored after this operation. For $x \in V(T_i)$, we let $\widetilde{G}_x$ denote the $t$-boundaried graph induced by the vertices in the bags of the subtree of $T_i$ rooted at $x$. That is, formally,

$$G_x := G\left[W_x \cup \bigcup\nolimits_{y \text{ descendant of } x} W_y\right] \text{ and } \widetilde{G}_x := (G_x, W_x)\,,$$

where the boundary $bd(G_x) = W_x$ is of size $t \leqslant d + h$. Then $G_r = G[Y_i \cup N(Y_i)]$. Note that the treedepth of $G_x$ is at most $d + |W_x \cap S| \leqslant d + h$.

Recall that $\Pi$ has FII either on general graphs or on bounded treedepth graphs. Using Lemma 3.5, for each $x \in V(T_i)$, there exists a representative $\Lambda(x) \in \mathcal{R}(d+h, d+h)$ of $\widetilde{G}_x$ which is an induced subgraph of $G_x$ and $bd(\Lambda(x)) = bd(G_x)$. Replacing $\widetilde{G}_x$ by $\Lambda(x)$ hence does not increase the treedepth. Furthermore, $|\Lambda(x)| \leqslant M := \rho(d+h, d+h)$ which is a constant. Let $\mu(x) = \Delta_t(\widetilde{G}_x, \Lambda(x))$.

Our task is to find $\Lambda(r)$ and $\mu(r)$ which we will calculate in a bottom-up manner along $T_i$ in $\mathcal{O}(|Y_i|)$ time as follows. If $y \in V(T_i)$ is a leaf node then these values can be computed in constant time. Let $x \in V(T_i)$ be a node with exactly one child $y$ whose $\Lambda$ and $\mu$ values are known. Consider the $t$-boundaried graph $\widetilde{G}'_x$ where $t \leqslant d+h$ and

$$G'_x := (G_x \ominus_{W_y} G_y) \oplus_{W_y} \Lambda(y) \text{ with } bd(G'_x) = W_x.$$

We claim that $\widetilde{G}'_x \equiv_t \widetilde{G}_x$. To prove this, we need to demonstrate that for all $t$-boundaried graphs $\widetilde{G}$ and all $\xi \in \mathbf{N_0}$,

$$(\widetilde{G}'_x \oplus_{W_x} \widetilde{G}, \xi) \in \Pi \text{ if and only if } (\widetilde{G}_x \oplus_{W_x} \widetilde{G}, \xi - \mu') \in \Pi,$$

where $\mu' = \Delta_t(\widetilde{G}_x, \widetilde{G}'_x)$ is to be specified. Now

$$\begin{aligned}(\widetilde{G}'_x \oplus_{W_x} \widetilde{G}, \xi) \in \Pi \text{ iff } & ((\widetilde{G}_x \ominus_{W_y} G_y) \oplus_{W_y} \Lambda(y)) \oplus_{W_x} \widetilde{G}, \xi) \in \Pi \\ \text{iff } & ((\widetilde{G}_x \oplus_{W_x} \widetilde{G}) \ominus_{W_y} G_y) \oplus_{W_y} \Lambda(y), \xi) \in \Pi \\ \text{iff } & ((\widetilde{G}_x \oplus_{W_x} \widetilde{G}) \ominus_{W_y} G_y) \oplus_{W_y} \widetilde{G}_y, \xi - \mu(y)) \in \Pi,\end{aligned}$$

where the last step follows because of $\Lambda(y) \equiv_{t'} \widetilde{G}_y$, where $t'$ is either $t' = t+1 \leqslant d+h$ in case $x$ is a forget node or $t' = t-1$ in case it is an introduce node. Since

$$(\widetilde{G}_x \oplus_{W_x} \widetilde{G}) \ominus_{W_y} G_y) \oplus_{W_y} \widetilde{G}_y \;=\; \widetilde{G}_x \oplus_{W_x} \widetilde{G},$$

this proves our claim. In fact, we have that $\mu' = \mu(y)$.

Observe that $G'_x$ is of *constant* size, bounded from above by $M + |W_x| \leqslant M + d + h = \mathcal{O}(1)$. Since $\Lambda(y)$ is an induced subgraph of $G_y$, it follows that $G'_x$ is an induced subgraph of $G_x$ and therefore has treedepth at most $d+h$. Then we can find in constant time the associated representative $\widetilde{R} \in \mathcal{R}(d+h, d+h)$ of $\widetilde{G}'_x$. We set $\Lambda(x) := \widetilde{R}$ and $\mu(x) := \mu' + \Delta_t(\widetilde{G}'_x, \widetilde{R})$. Note that the total time spent at node $x$ to generate these values is a constant.

Lastly, consider the case when $x \in V(T_i)$ has exactly two children $y_1$ and $y_2$ whose $\Lambda$ and $\mu$ values are known. Since our tree-decomposition is nice, we have $W_{y_1} = W_x = W_{y_2}$ and therefore $bd(G_{y_1}) = bd(G_{y_2}) = W_x$. Take the $t$-boundaried graph $\widetilde{G}''_x$ where $t \leqslant d+h$ and

$$G''_x := \Lambda(y_1) \oplus_{W_x} \Lambda(y_2) \text{ with } bd(G''_x) = W_x.$$

Similarly as in the previous case, one can show that for all graphs $\widetilde{G}$ and all $\xi \in \mathbf{N_0}$,

$$(\widetilde{G}''_x \oplus_{W_x} \widetilde{G}, \xi) \in \Pi \text{ if and only if } (\widetilde{G}_x \oplus_{W_x} \widetilde{G}, \xi - \mu'') \in \Pi,$$

where $\mu'' = \mu(y_1) + \mu(y_2)$. The graph $G''_x$ has size at most $2M$ which is a constant. One can therefore, again in constant time, calculate a representative $\widetilde{R} \in \mathcal{R}(d+h, d+h)$ of $\widetilde{G}''_x$. We set $\Lambda(x) := \widetilde{R}$ and $\mu(x) := \mu'' + \Delta_t(\widetilde{G}''_x, \widetilde{R})$.

To summarize, our proof shows that one can, independently for each $i \in \{1, \ldots, \ell\}$, in time $\mathcal{O}(|T_i|) = \mathcal{O}(|Y_i|)$ obtain $\Lambda(r)$ and $\mu(r)$ (where $r$ is the root of the tree-decomposition $\mathcal{T}_i$ for $G[Y_i \cup N(Y_i)]$) with the following properties: for all graphs $\widetilde{G}$ and all $\xi \in \mathbf{N_0}$,

$$(\widetilde{G}_r \oplus \widetilde{G}, \xi) \in \Pi \text{ if and only if } (\Lambda(r) \oplus \widetilde{G}, \xi + \mu(r)) \in \Pi.$$

Let $\mu_i := \mu(r)$ and $Y'_i := V(\Lambda(r)) \setminus Y_0$ be the chosen replacement of the cluster $Y_i$. Then $G[Y'_i]$ is an induced subgraph of $G[Y_i]$ of constant size, and the neighborhood of $Y'_i$ inside $Y_0$ is untouched. It immediately follows that $\mathbf{td}(G[Y'_i]) \leqslant \mathbf{td}(G[Y_i]) \leqslant d$ as claimed, too.

Finally, let $G' := G[Y_0 \cup Y'_1 \cup \cdots \cup Y'_\ell]$ and $\xi' := \xi + \mu_1 + \cdots + \mu_\ell$. The equivalence of the instances $(G, \xi)$ and $(G', \xi')$ of $\Pi$ then immediately follows from the safety of the protrusion replacement Rule 3.6. $\square$

With the lemmas at hand we can now prove the main theorem of this section.

*Proof of Theorem 4.1.* Given an instance $(G, \xi)$ of $\Pi$ with $G \in \mathcal{K}$, we calculate a $2^d$-approximate modulator $S$ using Lemma 4.2. Using the algorithm outlined in the proof of Lemma 4.6, we compute the decomposition $Y_0 \uplus Y_1 \uplus \cdots \uplus Y_\ell$. Each cluster $Y_i, 1 \leqslant i \leqslant \ell$, forms a protrusion with boundary size $|N(Y_i)| \leqslant 2^{d+1} + 2f(2^d) =: h$ and treedepth (and thus treewidth) at most $d$.

Applying Lemma 4.7 now yields an equivalent instance $(G', \xi')$ with $|V(G')| = |Y_0| + \sum_{i=1}^{\ell} |Y'_i|$ vertices, where $Y'_i$ denote the clusters obtained through applications of the reduction rule. This quantity is at most $\mathcal{O}(|S|) + \ell \cdot \rho(d+h, d+h) = \mathcal{O}(|S|)$ by Lemma 4.6 (3). As $G'$ is an induced subgraph of $G$, the above implies that $|V(G')| + |E(G')| = \mathcal{O}(|S|)$ by the degeneracy of $G$. $\square$

Finally, note that if the class $\mathcal{K}$ in the theorem is hereditary (which will be the case of our applications of Theorem 4.1), it also holds $G' \in \mathcal{K}$.

### *4.1. Extension to larger graph classes*

We can extend our result to classes of graphs that are *nowhere dense*, which present a wider framework than classes of bounded expansion.

**Definition 4.8** (Nowhere dense [31, 32])**.** A graph class $\mathcal{K}$ is *nowhere dense* if for all $r \in \mathbf{N_0}$ it holds that $\omega(\mathcal{K} \,\triangledown\, r) < \infty$.

In the above definition we use the natural extension of $\omega$ to classes of graphs via $\omega(\mathcal{K}) = \sup_{G \in \mathcal{K}} \omega(G)$. Note that nowhere dense classes are closed under taking shallow minors in the sense that $\mathcal{K} \,\triangledown\, r$ is nowhere dense if $\mathcal{K}$ is, albeit with a different bound on the clique size of $r$-shallow minors.

We claim the following kernelization result for nowhere dense classes, which in particular applies to all problems listed below in Section 5.

**Theorem 4.9.** *Let a class $\mathcal{K}$ be nowhere dense and let $d \in \mathbf{N_0}$ be a constant. Let $\Pi \in \mathfrak{P}$. There exists an algorithm that takes as input $(G, \xi) \in \mathcal{K} \times \mathbf{N_0}$ and, in time $\mathcal{O}(|G|^{1+\varepsilon})$ for every $\varepsilon > 0$, outputs $(G', \xi')$ such that*

1. *$(G, \xi) \in \Pi$ if and only if $(G', \xi') \in \Pi$;*
2. *$G'$ is an induced subgraph of $G$; and*
3. *$|G'| = \mathcal{O}(|S|^{1+\varepsilon})$, where $S$ is an optimal treedepth-$d$ modulator of $G$.*

Here we use the nowhere dense variant of Lemma 4.2 to obtain an approximate treedepth modulator in almost linear time. The proof of 4.9 follows analogously to the proof of 4.1, while replacing Lemma 4.3 with Lemma 4.11 (see below) and using the following property of nowhere dense classes:

**Proposition 4.10** ([7], also [6, Section 5.4])**.** *Let $\mathcal{G}$ be a nowhere dense graph class. Then for every $\alpha > 0$ and every $r \in \mathbf{N_0}$ there exists $n_{\alpha,r} \in \mathbf{N_0}$ such that for every $G \in \mathcal{G}$ with $|G| > n_{\alpha,r}$ it holds that $\nabla_r(G) \leqslant |G|^\alpha$.*

We need additional notation. For a graph class $\mathcal{G}$ and an integer $p$ we let $\mathcal{G}_{\leqslant p} := \{H \in \mathcal{G} \mid |H| \leqslant p\}$ denote those graphs of $\mathcal{G}$ which have at most $p$ vertices. We shortly write $G_{\leqslant p}$ for $(G \triangledown 0)_{\leqslant p}$.

**Lemma 4.11.** *Let $G = (X, Y, E)$ be a bipartite graph, and $p \geqslant \nabla_1(G_{\leqslant |X|^2})$. Then there are at most*

1. *$2p \cdot |X|$ vertices in $Y$ with degree greater than $\omega(G \triangledown 1)$; and*
2. *$(2p)^{\omega(G \triangledown 1)} \cdot |X|$ subsets $X' \subseteq X$ such that $X' = N(u)$ for some $u \in Y$.*

*Proof.* We construct a sequence of graphs $G_0 := G, G_1, \ldots, G_\ell$ in the same way as in the proof of Lemma 4.3. Recall that $G_i \in G \triangledown 1$ for $1 \leqslant i \leqslant \ell$, and so $\omega(G_\ell[X]) \leqslant \omega(G \triangledown 1)$, in particular. Furthermore, since every step $i$ of the sequence adds an edge to $G_i[X]$, we have $\ell < |X|^2/2$ and, consequently, $G_\ell[X]$ results by contracting at most $|X|^2/2$ vertices from $Y$ and so $G_\ell[X] \in G_{\leqslant |X|^2} \triangledown 1$. Then $G_\ell[X]$ is actually $2p$-degenerate and the first claim follows in exactly the same way as in 4.3.

For the second claim, consider again the set $Y' = Y \cap V(G_\ell)$. The neighborhood of every vertex $v \in Y'$ induces a clique in $G_\ell[X]$, as in Lemma 4.3. We additionally need a strengthening of Proposition 2.13:

Assume a graph $H$ and $v \in V(H)$ of degree $d$. Then the number of cliques in $H$ containing $v$ is clearly at most $\sum_{i=1}^{\omega(H)-1} \binom{d}{i} \leqslant d^{\omega(H)-1}$. If $H$ is $d$-degenerate, the overall number of cliques in $H$ is thus at most $d^{\omega(H)-1} \cdot |H|$. In our case of $H = G_\ell[X]$, there are at most $(2p)^{\omega(G \triangledown 1)-1} \cdot |X|$ possible cliques in $G_\ell[X]$. This quantity accounts for all possible distinct neighborhoods of vertices of $Y'$ in $X$, and summing with at most $\ell \leqslant 2p \cdot |X|$ neighborhoods of the vertices of $Y \setminus V(G_\ell)$ we get (with a large margin) the bound in the second claim. $\square$

The following two corollaries are analogues of Corollaries 4.4 and 4.5 and will be used in a similar fashion.

**Corollary 4.12.** *Let $\mathcal{K}$ be a nowhere dense graph class, and fix any $\varepsilon > 0$ and $\delta \in \mathbf{N_0}$. Let $q = \omega(\mathcal{K} \triangledown (\delta+1)) < \infty$. There exists $n_0 \in \mathbf{N_0}$, depending on $\mathcal{K}$ and $\varepsilon, \delta$, such that the following holds for every $G \in \mathcal{K}$ and $S \subseteq V(G)$, $|S| > n_0$: If $C_1, \ldots, C_s$ are disjoint connected subgraphs of $G - S$ satisfying $diam(C_i) \leqslant \delta$ and $|N_S(C_i)| > q$ for $i = 1, \ldots, s$, then $s \leqslant |S|^{1+\varepsilon}$.*

*Proof.* We construct an auxiliary bipartite graph $\bar{G}$ with partite sets $S$ and $Y = \{C_1, \ldots, C_s\}$. There is an edge between $C_i$ and $x \in S$ iff $x \in N_S(C_i)$. As in Corollary 4.4, we know that $\bar{G}$ is a depth-$\delta$ shallow minor of $G$ with branch sets $C_i, 1 \leqslant i \leqslant s$, and, for any $F \in \bar{G} \triangledown 1$, it is moreover $F \in G \triangledown (\delta+1)$. In particular, $\omega(\bar{G} \triangledown 1) \leqslant \omega(G \triangledown (\delta+1)) \leqslant q$. Though, we will also need the following small refinement of the previous fact:

Clearly, there exists a connected subgraph $C_i' \subseteq C_i$ such that $N_S(C_i') = N_S(C_i)$, $diam(C_i') \leqslant 2\delta$ and $|C_i'| \leqslant diam(C_i) \cdot |N_S(C_i)| + 1 < 2\delta|S|$—simply take a vertex $w \in V(C_i)$ together with shortest paths from $w$ to selected neighbors of $N_S(C_i)$ in $C_i$. Hence it holds for any $F \in \bar{G}_{\leqslant |S|^2} \triangledown 1$ that $F \in G_{\leqslant m} \triangledown (2\delta+1)$ where $m = |S|^2 \cdot 2\delta|S| = 2\delta|S|^3$.

Then, using also Proposition 4.10, $\nabla_1\big(\bar{G}_{\leqslant |S|^2}\big) \leqslant \nabla_{2\delta+1}\big(G_{\leqslant m}\big) \leqslant m^\alpha$ for any $\alpha > 0$ and all sufficiently large $|G|$ and $m$. We choose $\alpha = \varepsilon/4$. By the first claim of Lemma 4.11, for $p = m^\alpha$, we get that

$$s \leqslant 2p|S| = 2\big(2\delta|S|^3\big)^{\varepsilon/4} \cdot |S| < |S|^{1+\varepsilon}$$

whenever $|S|$ is sufficiently large. □

**Corollary 4.13.** *Let $\mathcal{K}$ be a nowhere dense graph class, and fix any $\varepsilon > 0$ and $\delta \in \mathbf{N_0}$. There exists $n_0 \in \mathbf{N_0}$, depending on $\mathcal{K}$ and $\varepsilon, \delta$, such that for every $G \in \mathcal{K}$ and $S \subseteq V(G)$, $|S| > n_0$, the following holds: If $\mathcal{C}_1, \ldots, \mathcal{C}_t$ are sets of connected components of $G - S$ such that*

1. *for all $C, C' \in \bigcup_i \mathcal{C}_i$ it holds that $C, C' \in \mathcal{C}_j$ for some $j$ if and only if $N_S(C) = N_S(C')$, and*
2. *for all $C \in \bigcup_i \mathcal{C}_i$, $diam(C) \leqslant \delta$,*

*then $t \leqslant |S|^{1+\varepsilon}$.*

*Proof.* Let $q = \omega(\mathcal{K} \triangledown (\delta+1)) < \infty$. As in the proof of Corollary 4.12, we construct a bipartite graph $\bar{G}$ with partite sets $S$ and $Y = \{C_1, \ldots, C_r\}$, where the vertices $C_j$, $1 \leqslant j \leqslant r$, represent the connected components in $\bigcup_i \mathcal{C}_i$ and $C_j$ has an edge to $x \in S$ iff $x \in N_S(C_j)$. As before, it holds for any $F \in \bar{G}_{\leqslant |S|^2} \triangledown 1$ that $F \in G_{\leqslant m} \triangledown (2\delta+1)$ where $m = 2\delta|S|^3$, and consequently $\nabla_1\big(\bar{G}_{\leqslant |S|^2}\big) \leqslant \nabla_{2\delta+1}\big(G_{\leqslant m}\big) \leqslant m^\alpha$ for any $\alpha > 0$ and all sufficiently large $|G|$ and $m$.

We now choose $\alpha = \varepsilon/(4q)$ and apply the second claim of Lemma 4.11;

$$\begin{aligned} t &\leqslant |\{S' \subseteq S \mid \exists C_i \in Y : N(C_i) = S'\}| \\ &\leqslant (2m^\alpha)^{\omega(\bar{G} \triangledown 1)} \cdot |S| \leqslant 2^q m^{\alpha q} \cdot |S| \\ &= 2^q m^{\varepsilon/4} \cdot |S| = 2^q (2\delta)^{\varepsilon/4} |S|^{3\varepsilon/4} \cdot |S| < |S|^{1+\varepsilon} \end{aligned}$$

whenever $|S|$ is sufficiently large. $\square$

We are now ready to prove the theorem. First, the following generalization of Lemma 4.6 follows easily using the above two corollaries.

**Lemma 4.14.** *Let $\mathcal{K}$ be a nowhere dense graph class, and fix any $\varepsilon > 0$ and $d \in \mathbf{N_0}$ ($d$ a constant). Let $q = \omega(\mathcal{K} \mathbin{\nabla} 2^d) < \infty$. Assume any $G \in \mathcal{G}$ and $S \subseteq V(G)$ a set of vertices such that $\mathbf{td}(G-S) \leqslant d$. There is an algorithm that partitions $V(G)$ in time linear in $|G|$ into sets $Y_0 \uplus Y_1 \uplus \cdots \uplus Y_\ell$ such that the following holds:*

1. *$S \subseteq Y_0$ and $|Y_0| = \mathcal{O}(|S|^{1+\varepsilon})$;*
2. *for $1 \leqslant i \leqslant \ell$, $Y_i$ induces a set of connected components of $G - Y_0$ that have the same neighborhood in $Y_0$ of size at most $2^{d+1} + q$; and*
3. *$\ell \leqslant \mathcal{O}(|S|^{1+\varepsilon})$.*

*Proof.* We use the same algorithm as in the proof of Lemma 4.6; setting the size of a large neighborhood to $q+1$ in accordance with the bound in Corollary 4.12. This proves the first two claims, provided $|S|$ is sufficiently large. The third claim then follows from the conclusion of Corollary 4.13. If, on the other hand, $|S|$ is bounded from above by a constant, then the claims follow from any trivial estimates; e.g., $s \leqslant |S|^2$ in place of Corollary 4.12 and $t \leqslant |S|^{q+1}$ in place of Corollary 4.13. $\square$

*Proof of Theorem 4.9.* The proof now proceeds in exactly the same way as that of Theorem 4.1. $\square$

## 5. FII and Structural Parameterization

Theorems 4.1 and 4.9, developed in the previous section, allow us to prove linear-kernelization results for many graph problems on classes of bounded expansion, and nearly-linear-kernelization results on nowhere dense graph classes, when the parameter is the size of a modulator to constant treedepth. Let us remind the reader that Theorems 4.1 and 4.9 apply to problems in the class $\mathfrak{P}$, where $\mathfrak{P}$ denotes the set of all graph problems that have FII on general graphs or, for each $p \in \mathbf{N_0}$, in the class of graphs of treedepth at most $p$. In this section we will present several classes of such problems, while postponing two more involved problems for Section 6.

Firstly, there is the large class of problems which have FII on general graphs. For these problems we immediately get the following:

**Corollary 5.1.** *The following graph problems have finite integer index, and hence linear ($\mathcal{O}(s)$) kernels in any hereditary graph class of bounded expansion, when the parameter is the size $s$ of a modulator to constant treedepth:*

Dominating Set, $r$-Dominating Set, Efficient Dominating Set, Connected Dominating Set, Vertex Cover, Hamiltonian Path, Hamiltonian Cycle, Connected Vertex Cover, Independent Set, Feedback Vertex Set, Edge Dominating Set, Induced Matching, Chordal Vertex Deletion, Odd Cycle Transversal, Induced $d$-Degree Subgraph, Min Leaf Spanning Tree, Max Full Degree Spanning Tree.

*Furthermore, under the same parameter $s$ the listed problems admit kernels of near-linear size $\mathcal{O}(s^{1+\varepsilon})$ for every $\varepsilon > 0$ in any hereditary nowhere dense graph class.*

For a more comprehensive list of problems that have FII in general graphs (and hence fall under the purview of Theorems 4.1 and 4.9), see [1].

The second class of problems are those which do not have FII in general graphs (see [33]), but only when restricted to graphs of bounded treedepth. Here we present four such problems.

**Lemma 5.2.** *Let $\mathcal{D}$ be a graph class of bounded treedepth. Then the problems* Longest Path, Longest Cycle, Exact $s,t$-Path, *and* Exact Cycle *have FII in $\mathcal{D}$.*

*Proof.* Let $\Pi$ be any one of the mentioned problems, and let $d, t$ be constants such that all graphs in $\mathcal{D}$ have treedepth $\leqslant d$. Consider the class $\mathcal{G}_t$ of all $t$-boundaried graphs, and let $T = \{0, 1, \ldots, t\}$.

We define a *configuration* of $\Pi$ with respect to $\mathcal{G}_t$ as a multiset

$$C = \{(s_1, d_1, t_1), \ldots, (s_p, d_p, t_p)\}$$

of triples from $(T \times \mathbf{N_0} \times T)$. We say a $t$-boundaried graph $\widetilde{G} \in \mathcal{G}_t$ *satisfies* the configuration $C$ if there exists a set of (distinct) paths $P_1, \ldots, P_p$ in $G$ such that

- $s_i, t_i$ can only be endvertices of $P_i$, $V(P_i) \cap bd(G) \subseteq \{s_i, t_i\}$, and $|P_i| = d_i$, for $1 \leqslant i \leqslant p$,
- $V(P_i) \cap V(P_j) \subseteq bd(G)$ for $1 \leqslant i < j \leqslant p$, and
- $V(P_i) \cap V(P_j) \cap V(P_k) = \emptyset$ for $1 \leqslant i < j < k \leqslant p$.

Note that, for simplicity, we identify the boundary vertices in $bd(G)$ with their labels $1, \ldots, t$ from $T$. Moreover, $s_i, t_i$ can take the value 0 which is not contained in $bd(G)$: semantically these tuples describe paths which intersect the boundary of $G$ at only one or no vertex. Another special case are tuples with $s_i = t_i$ and $d = 0$: those describe single vertices of the boundary. In short, a graph satisfies a configuration if it contains internally non-intersecting paths of length and endvertices prescribed by the tuples of the configuration, and no three of the paths are prescribed to have the same endvertex (hence some configurations are not satisfiable at all).

The *signature* $\sigma[\widetilde{G}]$ of a graph $\widetilde{G} \in \mathcal{G}_t$ is a function from the configurations into $\{0,1\}$ where $\sigma[\widetilde{G}](C) = 1$ iff $\widetilde{G}$ satisfies $C$. We define:

$$\widetilde{G}_1 \simeq_\sigma \widetilde{G}_2 \iff \sigma[\widetilde{G}_1] \equiv \sigma[\widetilde{G}_2] \text{ for } \widetilde{G}_1, \widetilde{G}_2 \in \mathcal{G}_t.$$

We claim that the equivalence relation $\simeq_\sigma$ is a refinement of $\equiv_{\Pi,t}$. We provide only a sketch for $\Pi =$ LONGEST PATH, the proofs for the other problems work analogously. To this end we assume the contrary, that $\sigma[\widetilde{G}_1] \equiv \sigma[\widetilde{G}_2]$ while $\widetilde{G}_1 \not\equiv_t \widetilde{G}_2$. Up to symmetry, this means that for all integers $c$ there exists a graph $\widetilde{G}_3 \in \mathcal{G}_t$ such that $(\widetilde{G}_1 \oplus \widetilde{G}_3, \ell) \in \Pi$ but $(\widetilde{G}_2 \oplus \widetilde{G}_3, \ell + c) \notin \Pi$. We choose $c = 0$ and show the contradiction. Thus the graph $\widetilde{G}_1 \oplus \widetilde{G}_3$ contains a path $P$ of length $\ell$ but $\widetilde{G}_2 \oplus \widetilde{G}_3$ does not.

Using the implicit order given through the vertex order of $P$ we sort the subpaths of $P$ contained in $P \cap G_1$ and so obtain a sequence of paths $P_1, \ldots, P_q \subseteq G_1$, each with at most two vertices – the ends, in $bd(G_1)$. By identifying each subpath $P_i$ with the tuple $(s_i, d_i, t_i)$ where $d_i = |P_i|$ and $s_i$ is the label of the start of $P_i$ in $bd(G_1)$ (or 0 if $s_i \notin bd(G_1)$) and $t_i$ the label of the end of $P_i$ in $bd(G_1)$ (ditto), we obtain a configuration $C_P = \{(s_1, d_1, t_1), \ldots, (s_q, d_q, t_q)\}$. Now, $\widetilde{G}_1$ satisfies $C_P$ by the definition. Since $\sigma[\widetilde{G}_1](C_P) = \sigma[\widetilde{G}_2](C_P)$, there exists a set of paths $Q_1, \ldots, Q_q \subseteq G_2$ witnessing that $\widetilde{G}_2$ satisfies $C_P$. But then $Q_1, \ldots, Q_q$ together with $P \cap G_3$ form a path $Q$ of length $\ell$ in $\widetilde{G}_2 \oplus \widetilde{G}_3$, a contradiction.

Second, although $\simeq_\sigma$ is generally of infinite index, we claim that for every $t$, only a finite number of equivalence classes of $\simeq_\sigma$ carry a representative of treedepth $\leqslant d$, and hence $\simeq_\sigma$ is of finite index when restricted to graphs from $\mathcal{D}$. This is rather easy since graphs of treedepth $\leqslant d$ do not contain paths of length $2^d - 1$ or longer, and so a graph $\widetilde{G} \in \mathcal{D}_t$ can satisfy a configuration $C = \{(s_1, d_1, t_1), \ldots, (s_p, d_p, t_p)\}$ only if $d_i \in \{0, 1, \ldots, 2^d - 2\}$ for $1 \leqslant i \leqslant p$. Recall, each boundary vertex label occurs at most twice among $s_1, t_1, \ldots, s_p, t_p$ in a satisfiable configuration. Hence only finitely many such configurations $C$ can be satisfied by a graph from $\mathcal{D}_t$, and consequently, finitely many function values of $\sigma[\widetilde{G}]$ are nonzero for any $\widetilde{G} \in \mathcal{D}_t$ and the number of the nonempty classes of $\simeq_\sigma$ restricted to $\mathcal{D}_t$ is finite. $\square$

For these problems we can, again using Theorems 4.1 and 4.9, conclude the following:

**Corollary 5.3.** *The problems* LONGEST PATH, LONGEST CYCLE, EXACT $s,t$-PATH, *and* EXACT PATH *have linear kernels in any hereditary graph class of bounded expansion, with the size $s$ of a modulator to constant treedepth as the parameter. Furthermore, under the same parameter $s$ the listed problems admit kernels of near-linear size $\mathcal{O}(s^{1+\varepsilon})$ for every $\varepsilon > 0$ in any hereditary and nowhere dense graph class.*

The third class of problems we consider are the problems associated with the well known graph width measures branchwidth, pathwidth and treewidth. The problems are defined as follows: The BRANCHWIDTH (PATHWIDTH, TREEWIDTH) problem is, given a graph $G$ and an integer $k$, to decide whether $G$ has branchwidth (or pathwidth, treewidth respectively) at most $k$.

First thing about these problems is that they do not have FII on general graphs. For pathwidth and treewidth this can be easily proved using the fact that the complete graph on $n$ vertices, $K_n$, has pathwidth and treewidth $n-1$.

**Proposition 5.4.** *The problems* PATHWIDTH *and* TREEWIDTH *do not have FII on the class of all graphs.*

*Proof.* For $n, t \in \mathbf{N_0}$, $n > t$, let $\widetilde{K}_n = (K_n, bd(K_n))$ be the $t$-boundaried complete graph with $n$ vertices. We claim that $\widetilde{K}_m \not\equiv_{\mathbf{pw},t} \widetilde{K}_n$ and $\widetilde{K}_m \not\equiv_{\mathbf{tw},t} \widetilde{K}_n$ for every $m, n \in \mathbf{N_0}$ with $t < m < n$. This shows that neither $\equiv_{\mathbf{pw},t}$ nor $\equiv_{\mathbf{tw},t}$ is finite and concludes the proof of the theorem.

Let $\widetilde{H}_1 = \widetilde{K}_m$ and $\widetilde{H}_2 = \widetilde{K}_n$. Then, $\mathbf{pw}(\widetilde{K}_m \oplus \widetilde{H}_1) = m-1$ and $\mathbf{pw}(\widetilde{K}_n \oplus \widetilde{H}_1) = n-1$ but $\mathbf{pw}(\widetilde{K}_m \oplus \widetilde{H}_2) = n-1$ and $\mathbf{pw}(\widetilde{K}_n \oplus \widetilde{H}_2) = n-1$, as required. The proof for $\equiv_{\mathbf{tw},t}$ is identical. □

A similar proof for BRANCHWIDTH can be obtained using the well-known fact that the branchwidth of $K_n$ is $\mathbf{bw}(K_n) = \lceil 2/3 \cdot n \rceil$.

The fact that none of the above problems has FII on general graphs motivates us to take a closer look at restricted graph classes, which still provide us with enough power to apply the protrusion replacement machinery. We start with the relatively easy case of BRANCHWIDTH, and postpone the, significantly more difficult, problems PATHWIDTH and TREEWIDTH to Section 6.

**Lemma 5.5.** *Let $\mathcal{B}$ be a graph class of bounded branchwidth. Then* BRANCHWIDTH *has FII in $\mathcal{B}$.*

*Proof.* Let $\mathcal{G}_t$ be the class of all $t$-boundaried graphs. Let $\mathcal{X}^w$ denote the set of minor-minimal graphs of branchwidth greater than $w$ (we will see that $\mathcal{X}^w$ is finite for every $w$ but that is not important for now). That is, $G \in \mathcal{X}^w$ if and only if the branchwidth of $G$ is $> w$ but every proper minor of $G$ has branchwidth $\leqslant w$; $G$ is an "obstruction" to branchwidth $w$. We also say that, for a graph $G$ and $A \subseteq V(G)$, a graph $H$ is an *$A$-restricted* minor of $G$ if $H$ can be obtained from $G$ by only deleting vertices of $A$ and contracting or deleting edges with both ends in $A$. Let $\mathcal{X}^w_{*t} \subseteq \mathcal{G}_t$ be the "$t$-boundaried fragments" of members of $\mathcal{X}^w$, up to isomorphism, i.e.

$$\widetilde{F} \in \mathcal{X}^w_{*t} \iff \exists \widetilde{F}' \text{ s.t. a } bd(\widetilde{F})\text{-restricted minor of } \widetilde{F} \oplus \widetilde{F}' \text{ belongs to } \mathcal{X}^w.$$

Let $\Pi$ be the problem BRANCHWIDTH. The framework of the proof is very similar to that of Lemma 5.2; members of $\mathcal{X}^w_{*t}$ play the role of configurations of $\Pi$ and a signature is a subset of $\mathcal{X}_{*t} := \bigcup_w \mathcal{X}^w_{*t}$. First, for a $t$-boundaried graph $\widetilde{G}$, the *signature* $\sigma[\widetilde{G}]$ is defined as the set of those $\widetilde{F} \in \mathcal{X}_{*t}$ such that $\widetilde{F}$ is a *rooted minor* of $\widetilde{G}$, meaning that $F$ is a minor of $G$ in such a way that the boundary $bd(F) = bd(G)$ is identical (not touched). It is routine to verify that if, informally, the same fragments of "branchwidth obstructions" exist in both $\widetilde{G}_1$ and $\widetilde{G}_2$, then they are equivalent. Formally;

$$\text{if } \sigma[\widetilde{G}_1] = \sigma[\widetilde{G}_2], \text{ then } \widetilde{G}_1 \equiv_{\Pi,t} \widetilde{G}_2 \text{ with } \Delta_{\Pi,t}(\widetilde{G}_1, \widetilde{G}_2) = 0.$$

Second, the equivalence relation $\simeq_\sigma$ on $\mathcal{G}_t$ defined by the same signature $\sigma$ is generally of infinite index, though, we claim that for every $b, t$, only a finite number of equivalence classes of $\simeq_\sigma$ carry a representative of branchwidth $\leqslant b$. This would follow if we proved that only finitely many elements of $\mathcal{X}_{*t}$ have branchwidth $\leqslant b$. The latter is a nontrivial statement, possible thanks to some fine properties of the "branchwidth obstructions" as proved in [34] (note that although the paper deals with matroids, its results apply to graph branchwidth obstructions as well since graph branchwidth equals branchwidth of the cycle matroid of the graph [35]). Precisely, besides finiteness of $\mathcal{X}^w$ for each $w$, we use [34, Lemma 4.1] which, in our terms, reads:

> If $\widetilde{H}, \widetilde{H}'$ are $k$-boundaried graphs such that $\widetilde{H} \oplus \widetilde{H}' \in \mathcal{X}^w$ and $k \leqslant w$, then $|E(H)| \leqslant g(k)$ or $|E(H')| \leqslant g(k)$, where $g(k) = (6^{k-1} - 1)/5$.

Assume now $\widetilde{F} \in \mathcal{X}_{*t}$ such that the underlying graph $F$ is of branchwidth $b$, and let $w_0 = b+g(t)+\binom{t}{2}$. Either, $\widetilde{F} \in \bigcup_{w<w_0} \mathcal{X}^w_{*t}$ which is a finite set, or there is $\widetilde{F}'$ and a $bd(\widetilde{F})$-restricted minor $F_1$ of $\widetilde{F} \oplus \widetilde{F}'$ such that $F_1 \in \mathcal{X}^w$ where $w \geqslant w_0$. Note that, in particular, the branchwidth of $\widetilde{F} \oplus \widetilde{F}'$ is greater than $w$, and so the branchwidth of $F$ is greater than $w - |E(F')|$ since branchwidth drops by at most one when deleting or contracting a single edge. If $|E(F')| \leqslant g(t)+\binom{t}{2}$, then the branchwidth of $F$ is greater than $w - |E(F')| \geqslant b+g(t)+\binom{t}{2} - g(t) - \binom{t}{2} = b$, a contradiction. Hence, $|E(F')| > g(t) + \binom{t}{2}$. Let $F_1 = \widetilde{H}_1 \oplus \widetilde{H}'_1$ such that $\widetilde{H}_1$ is a rooted minor of $\widetilde{F}$ and $\widetilde{H}'_1$ is a rooted minor of $\widetilde{F}'$. Since $F_1$ is a $bd(\widetilde{F})$-restricted minor of $\widetilde{F} \oplus \widetilde{F}'$ (meaning that one is allowed to contract or delete only edges with both ends in $bd(\widetilde{F})$), it holds $|E(\widetilde{F} \oplus \widetilde{F}')| - |E(F_1)| \leqslant \binom{t}{2}$ and so $|E(F')| - |E(H'_1)| \leqslant \binom{t}{2}$. Consequently, $|E(H'_1)| > g(t)$. The boundary of $\widetilde{H}'_1$ is of size $k \leqslant t$, and so [34, Lemma 4.1] can be applied to $F_1$. Therefore, we have $|E(H_1)| \leqslant g(k) \leqslant g(t)$ and $|E(F)| \leqslant g(t) + \binom{t}{2}$, and there are only finitely many such $t$-boundaried graphs without isolated vertices in $\mathcal{X}_{*t}$. □

Since branchwidth is bounded if, and only if, treewidth is bounded, and by Proposition 2.12e, we can now apply Theorems 4.1 and 4.9 to conclude that:

**Corollary 5.6.** *The problem* BRANCHWIDTH *has a linear kernel in any hereditary graph class of bounded expansion, with the size $s$ of a modulator to constant treedepth as the parameter. Furthermore, under the same parameter $s$ the problem admits a kernel of near-linear size $\mathcal{O}(s^{1+\varepsilon})$ for every $\varepsilon > 0$ in any hereditary and nowhere dense graph class.*

Somehow surprisingly, it is not at all easy to extend the arguments of Lemma 5.5 to the related problems PATHWIDTH and TREEWIDTH, since we do not have any direct analogue of the results of [34] for the other measures.

## 6. FII of Pathwidth and Treewidth

We dedicate this section to proving that the problems Pathwidth and Treewidth have FII on graphs of bounded pathwidth and treewidth, respectively. Compared to the path and cycle problems treated in Lemma 5.2 and the Branchwidth problem treated in Lemma 5.5, the proofs here are much more involved and use the notion of characteristics of path decompositions and tree decompositions, which have been introduced in [36]. Because the definition of these characteristics is quite technical and the properties we require have already been shown in [36], we will not provide a formal definition. Instead, we will only state the required properties and refer the reader to [36] for details and proofs.

The concept of a *characteristic* of a partial path decomposition of a graph—or equivalently the characteristic of a path decomposition of a boundaried graph—was introduced by Bodlaender and Kloks in [36, Definition 4.4]. Informally, the characteristic of a path decomposition $\mathcal{P}$ of $\widetilde{G}$ compactly represents all the information required to compute, for any $\widetilde{H}$, the ways $\mathcal{P}$ can be extended into a path decomposition of the graph $\widetilde{G} \oplus \widetilde{H}$. This information can then be used to compute the pathwidth of the graph $\widetilde{G} \oplus \widetilde{H}$. Importantly, the number of characteristics of path decompositions of width at most $w$ of any $t$-boundaried graph only depends on $t$ and $w$, but not on the the graph itself.

**Proposition 6.1** ([36, Lemma 4.1]). *Let $\widetilde{G}$ be a $t$-boundaried graph and $w$ an integer. Then the number of characteristics of path decompositions of width at most $w$ of $\widetilde{G}$ is bounded by a function of $t$ and $w$.*

For integer $w$, the *full set of (path decomposition) characteristics* of $\widetilde{G}$ of width at most $w$ (as defined in [36, Definition 4.6]), denoted by $\mathrm{FSCP}_w(\widetilde{G})$, is the set of all characteristics of path decompositions of $\widetilde{G}$ of width at most $w$. We denote by $\mathrm{FSCP}(\widetilde{G})$ the (possibly infinite) set $\bigcup_{w \in \mathbf{N_0}} \mathrm{FSCP}_w(\widetilde{G})$. Recall the definition of the projection of a path-decomposition, denoted by $|$, from Section 2.2.

**Proposition 6.2** ([36, Section 4.3]). *Let $\widetilde{H}$, $\widetilde{G}_1$ and $\widetilde{G}_2$ be $t$-boundaried graphs, and let $\mathcal{P}$ be a path decomposition of $\widetilde{G}_1 \oplus \widetilde{H}$. If the (unique) characteristic of $\mathcal{P}|G_1$ is in $\mathrm{FSCP}(\widetilde{G}_2)$, then there is a path decomposition of $\widetilde{G}_2 \oplus \widetilde{H}$ that has the same width as $\mathcal{P}$.*

*Proof sketch.* Since the proof relies on an informal understanding of the algorithm described in [36, Section 4.3], we will start with a brief description of this algorithm. Given a graph $G$, an integer $k$, and a path decomposition $\mathcal{P}$ of $G$ with width $l$, the algorithm described in [36, Section 4.3] decides whether $G$ has a path decomposition of width at most $k$ (and if so computes such a path decomposition). The algorithm uses a standard dynamic programming algorithm, as is the usual approach to solving problems on graphs of bounded pathwidth. That is, the algorithm computes a set of records (which is called a full set of characteristics in [36]) for each node of the path decomposition $\mathcal{P}$ in a left to right manner, i.e., starting at the "left" endpoint of the path decomposition, the algorithm computes such a full set of characteristics for each node from a full set of characteristics of its left neighbor in $\mathcal{P}$ until such a full set of characteristics is eventually computed for the "right" endpoint of $\mathcal{P}$. For a node $p$ of $\mathcal{P}$ let $G(p)$ be the subgraph of $G$ induced by all the vertices contained in bags of $\mathcal{P}$ that are to the left of $p$ in $\mathcal{P}$ (including the bag of $p$ itself). Then the full set of characteristics for $p$, denoted by $F(p)$, contains one characteristic for every (partial) path decomposition of $G(p)$ of width at most $k$. Informally, a characteristic is a compact representation of a partial path decomposition of $G(p)$ that contains sufficient information such that the algorithm will later be able to decide how the partial path decomposition represented by the characteristic can be extended to a partial path decomposition of $G(p')$ for the right neighbor $p'$ of $p$ in $\mathcal{P}$. The crucial point we will employ for our proof below is that the computations made after a certain bag $p$ of $\mathcal{P}$ to a characteristic in $F(p)$ only depends on the characteristic itself and the set of bags coming after and including $p$ in $\mathcal{P}$ but not on the set of bags coming before $p$. This is the usual behavior of a dynamic programming algorithm on a path decomposition and the algorithm given in [36] is no exception to that rule.

For the interested reader we will now give a brief and informal description of the characteristics defined in [36, Definition 4.4]. Let $\mathcal{P}'$ be a partial path decomposition of $G(p)$ and let $S$ be the set of vertices contained in the bag of $p$. The *interval model* of $\mathcal{P}'$ (as defined in [36, Definition 3.3]) is the sequence $(Z_1, \ldots, Z_q)$ of bags obtained from the projection of $\mathcal{P}'$ onto $S$ after deleting consecutive bags having the same content. Then a characteristic for $\mathcal{P}'$ as defined in [36, Definition 4.4] consists of the interval model of $\mathcal{P}'$ together with the "typical list of integers" for each bag of the interval model. Here, the "typical list of integers" for a bag $Z_i$ of the interval model is a compact representation of the list of integers given by the sizes of the bags in $\mathcal{P}'$, whose intersection with $S$ is equal to $Z_i$. For our argumentation given below the exact definition of the "typical list of integers" is not important, i.e., for us it is only important that the characteristic of $\mathcal{P}'$ is independent of the graph $G(p) \setminus S$ in the sense that it contains no information about particular vertices or edges in $G(p) \setminus S$ but only about the number of these vertices.

We are now ready to complete the proof of the proposition. For $i \in \{1, 2\}$, let $\mathcal{P}_i$ be any path decomposition of $\widetilde{G}_i$ such that the content of the last bag of $\mathcal{P}_i$ is $bd(G_i)$ and let $\mathcal{P}_3$ be any path decomposition of $\widetilde{H}$ such that the content of the first bag of $\mathcal{P}_3$ is $bd(H)$. Furthermore, for $i \in \{1, 2\}$, let $\mathcal{P}_{i,3}$ be the path decomposition of $\widetilde{G}_i \oplus \widetilde{H}$ obtained from $\mathcal{P}_i$ and $\mathcal{P}_3$ by appending the first bag of $\mathcal{P}_3$ to the last bag of $\mathcal{P}_i$, let $p_{i,3}$ be the bag of $\mathcal{P}_{i,3}$ that corresponds to the last bag of $\mathcal{P}_i$, and let $l_{i,3}$ be the last bag of $\mathcal{P}_{i,3}$.

Now assume that we run the algorithm described in [36, Section 4.3] on the path decomposition $\mathcal{P}_{i,3}$ and let $F(p_{i,3})$ and $F(l_{i,3})$ be the full set of characteristics of partial path decompositions computed at the node $p_{i,3}$ and the node $l_{i,3}$, respectively, of width at most the width of $\mathcal{P}$.

Then, by the definition of a full set of characteristics, we obtain that $F(p_{1,3})$ contains the characteristic of $\mathcal{P}|G_1$ and that $F(l_{1,3})$ contains the characteristic of $\mathcal{P}$. Moreover, the characteristic of $\mathcal{P}$ in $F(l_{1,3})$ is generated by the algorithm from the characteristic of $\mathcal{P}|G_1$ in $F(p_{1,3})$. By the assumptions of the Proposition, we have that the characteristic of $\mathcal{P}|G_1$ is contained in $\mathrm{FSCP}(\widetilde{G}_2)$ and hence also in $F(p_{2,3})$. Hence, because the path decompositions $\mathcal{P}_{1,3}$ and $\mathcal{P}_{2,3}$ are identical with respect to everything behind the nodes $p_{1,3}$ and $p_{2,3}$, respectively, we obtain that the characteristic of $\mathcal{P}$ is also contained in $F(l_{2,3})$, witnessing that $\widetilde{G}_2 \oplus \widetilde{H}$ has a path decomposition with the same width as $\mathcal{P}$. □

The above Proposition illuminates the usefulness of characteristics to show FII for the PATHWIDTH problem. In particular, it follows that if $\mathrm{FSCP}(\widetilde{G}_1) = \mathrm{FSCP}(\widetilde{G}_2)$, then $\widetilde{G}_1 \equiv_{\mathbf{pw},t} \widetilde{G}_2$, for all $t$-boundaried graphs $\widetilde{G}_1$ and $\widetilde{G}_2$. Hence, the full set of characteristics of a boundaried graph fully describes its equivalence class with respect to $\equiv_{\mathbf{pw},t}$. However, as mentioned above the full set of characteristics of a boundaried graph can be infinite. We will later show that if we consider FII with respect to a class $\mathcal{C}$ of graphs of bounded pathwidth, then it is sufficient to consider the set $\mathrm{FSCP}_{(\mathbf{pw}(\widetilde{G})+t)}(\widetilde{G})$ instead of $\mathrm{FSCP}(\widetilde{G})$ for every $t$-boundaried graph $\widetilde{G} = (G, bd(G))$ with $G \in \mathcal{C}$. Because $\mathbf{pw}(\widetilde{G})$ is bounded by a constant, the set of characteristics $\mathrm{FSCP}_{(\mathbf{pw}(\widetilde{G})+t)}$ is finite due to Proposition 6.1.

In the rest of this subsection we introduce characteristics for tree decompositions of boundaried graphs. All the explanations for characteristics of path decompositions transfer to characteristics of tree decompositions and we will not repeat them here. In [36, Definition 5.9] the authors define the *characteristic* of a tree decomposition of a boundaried graph. They show the following:

**Proposition 6.3** ([36, Remark below Lemma 5.3]). *Let $\widetilde{G}$ be a $t$-boundaried graph and $w$ an integer. Then the number of characteristics of tree decompositions of width at most $w$ of $\widetilde{G}$ is bounded by a function of $t$ and $w$.*

For an integer $w$, the *full set of (tree decomposition) characteristics* of $\widetilde{G}$ of width at most $w$ (as defined in [36, Definition 5.11]), denoted by $\mathrm{FSCT}_w(\widetilde{G})$, is the set of all characteristics of tree decompositions of $\widetilde{G}$ of width at most $w$. We denote by $\mathrm{FSCT}(\widetilde{G})$ the (possibly infinite) set $\bigcup_{w\in\mathbf{N_0}} \mathrm{FSCT}_w(\widetilde{G})$.

**Proposition 6.4** ([36, Section 5.3]). *Let $\widetilde{H}$, $\widetilde{G}_1$ and $\widetilde{G}_2$ be $t$-boundaried graphs, and let $\mathcal{T}$ be a tree decomposition of $\widetilde{G}_1 \oplus \widetilde{H}$. If the (unique) characteristic of $\mathcal{P}|G_1$ is in $\mathrm{FSCT}(\widetilde{G}_2)$, then there is a tree decomposition of $\widetilde{G}_2 \oplus \widetilde{H}$ that has the same width as $\mathcal{T}$.*

### *6.1.* PATHWIDTH *has FII on graphs of small pathwidth*

In this section we will make use of characteristics of path decompositions of boundaried graphs to show FII for the PATHWIDTH problem in a class of graphs of bounded pathwidth. In particular, we will show that the equivalence relation $\simeq_{\mathbf{pw},t}$ defined by

$$\widetilde{G}_1 \simeq_{\mathbf{pw},t} \widetilde{G}_2 \text{ if and only if } \mathrm{FSCP}_{(\mathbf{pw}(G_1)+t)}(\widetilde{G}_1) = \mathrm{FSCP}_{(\mathbf{pw}(G_2)+t)}(\widetilde{G}_2)$$

is a refinement of the equivalence relation $\equiv_{\mathbf{pw},t}$. The following lemma, which we believe to be interesting in its own right, is central to our proof.

**Lemma 6.5.** *Let $\widetilde{G}_1, \widetilde{G}_2$ be two $t$-boundaried graphs, $G = \widetilde{G}_1 \oplus \widetilde{G}_2$, and $\mathcal{P} = (P, \chi)$ be a path decomposition of $G$. Then there is a path decomposition $\mathcal{P}' = (P', \chi')$ of $G$ of the same width as $\mathcal{P}$ such that $\mathcal{P}'|G_1$ has width at most $\mathbf{pw}(G_1) + t$.*

*Proof.* If $\mathcal{P}|G_1$ has width at most $\mathbf{pw}(G_1) + t$, then $\mathcal{P}' := \mathcal{P}$ is the required path decomposition of $G$. Otherwise, there is a bag $p \in V(P)$ such that $|\chi(p) \cap V(G_1)| > \mathbf{pw}(G_1) + t + 1$, and we call such a bag $p$ a *bad bag* of $\mathcal{P}$. The next claim shows that we can eliminate the bad bags of $\mathcal{P}$ one by one without introducing new bad bags. Hence, we obtain the desired path decomposition $\mathcal{P}'$ from $\mathcal{P}$ by a repeated application of the following claim:

**Claim.** *There is a path decomposition $\mathcal{P}'' = (P'', \chi'')$ of $G$ of the same width as $\mathcal{P}$ such that the set of bad bags of $\mathcal{P}''$ is a proper subset of the set of bad bags of $\mathcal{P}$. Moreover, the bag $p$ is no longer a bad bag of $\mathcal{P}''$.*

For a subgraph $G'$ of $G$ and a bag $p'$ of $\mathcal{P}$, let $\chi_{G'}(p')$ be the set of vertices $\chi(p) \cap V(G')$ and let $S$ be a minimum separator between $\chi_{G_1}(p)$ and $bd(G_1)$ in the graph $G$. Since $bd(G_1)$ separates $\chi_{G_1}(p)$ from $bd(G_1)$ and is of cardinality at most $t$, we obtain that $|S| \leqslant t$. Let $W$ be the set of all vertices reachable from $\chi_{G_1}(p)$ in $G - S$, and let $\mathcal{P}_W = (P_W, \chi_W)$ be an optimal path decomposition of $G[W]$. Then, because $W \subseteq V(G_1)$, it follows that the width of $\mathcal{P}_W$ is at most the pathwidth of $G_1$.

To obtain the desired path decomposition $\mathcal{P}''$, where $p$ is not a bad bag anymore, we delete all vertices of $W$ from the bags of $\mathcal{P}$ and, instead, insert the path decomposition $\mathcal{P}_W$ between $p$ and an arbitrary neighbor of $p$ in $P$. To ensure Property P3 of a path decomposition for the vertices in $\chi(p) \setminus V(G_1)$, we add $\chi(p) \setminus V(G_1)$ to every bag of $\mathcal{P}_W$ in $\mathcal{P}''$. Furthermore, to cover the edges between $S$ and $W$ in $G$ we also need to add $S$ to $p$ and every bag of $\mathcal{P}_W$. Observe that after applying the above modifications the size of any bag that originated from $\mathcal{P}_W$ is at most $\mathbf{pw}(G_1) + 1 + t + |\chi(p) \setminus V(G_1)|$, which is at most the size of the original bad bag $p$ and hence bounded by the width of $\mathcal{P}$ plus one. Moreover, the sizes of all bags that originated from $\mathcal{P}$ did so far only decrease and in particular the intersection of the bad bag $p$ with $G_1$ is now exactly $S$. Since in the following we will do no further modifications to the bags originating from $\mathcal{P}_W$ and the bag $p$, we already obtain that the size of those bags is at most the width of $\mathcal{P}$ and $p$ is not a bad bag anymore.

Because $\chi(p)$ does not necessarily contain all vertices of $S$, this could potentially violate the Property P3 of a path decomposition. To get around this we will add a vertex $s \in S$ to every bag $p' \in V(P)$ in between $p$ and any bag containing $s$, i.e., we complete $\mathcal{P}''$ into a valid path decomposition in a minimal way. This completes the construction of $\mathcal{P}''$. As stated in the previous paragraph, neither $p$ nor any bag originating from $\mathcal{P}_W$ is a bad bag and moreover the sizes of these bags is at most the width of $\mathcal{P}$. Because all the vertices that we added or removed from bags originating from $\mathcal{P}$ are contained in $G_1$, it suffices to show that we never added more vertices to these bags than we removed. Suppose not, and let $p_2$ (not equal to $p$ and originating from $\mathcal{P}$) be a bag where we add more vertices than we remove. It follows that there is a bag $p_1 \in V(P)$ such that $p_2$ lies on the path from $p_1$ to $p$ in $\mathcal{P}$ and $|R| < |S'|$, where $R = \chi(p_2) \cap W$ and $S' = (\chi(p_1) \setminus \chi(p_2)) \cap S$. Note that in $\mathcal{P}|G_1[W \cup S']$ we have $\chi_{G_1[W\cup S']}(p_2) = R$. Because of Proposition 2.6 applied to $\mathcal{P}|G_1[W \cup S']$, $R$ separates $\chi_{G_1[W\cup S']}(p)$ from $S'$ in $G_1[W \cup S']$.

We claim that $S'' = (S \setminus S') \cup R$ is a separator between $\chi_{G_1}(p)$ and $bd(G_1)$. Since $|S''| < |S|$, this would contradict the minimality of $S$. Let $\Pi$ be a path between $\chi_{G_1}(p)$ and $bd(G_1)$. Since $\chi_{G_1}(p) \subseteq W \cup S$, $\Pi$ has to intersect $S$ in order to reach $bd(G_1)$. Let $s$ be the first vertex of $\Pi$ which intersects $S$ (note that the subpath from $\chi_{G_1}(p)$ to $s$ of $\Pi$ lies entirely in $W$). Either $s \in S \setminus S'$ and therefore $s \in S''$, or $s \in S'$ and the subpath from $\chi_{G_1}(p)$ to $s$ of $\Pi$ lies entirely in $W \cup S'$, and therefore $\Pi$ has to intersect $R \subseteq S''$ in order to reach $s$. It follows that $S''$ is indeed a separator between $\chi_{G_1}(p)$ and $bd(G_1)$, completing the proof. ☐

We note here that the bound for the pathwidth given in the above lemma is essentially tight. To see this, consider the complete bipartite graph $G$ that has $t$ vertices on one side (side $A$) and $t+1$ vertices on the other side (side $B$). Let $\widetilde{G}_1$ be the graph $G[A]$ with boundary $A$, let $\widetilde{G}_2$ be the graph $G$ with boundary $A$, and let $\mathcal{P}$ be any optimal path decomposition of $\widetilde{G}_1 \oplus \widetilde{G}_2 = G$. Then, because $G$ is a complete bipartite graph, whose smaller side is $A$, it holds that $\mathcal{P}$ contains a bag containing $A$. Consequently, $\mathbf{pw}(\mathcal{P}|G_1) = t - 1$ while $\mathbf{pw}(G_1) = 0$.

**Corollary 6.6.** *Let $\widetilde{G}_1$ and $\widetilde{G}_2$ be two $t$-boundaried graphs and $G = \widetilde{G}_1 \oplus \widetilde{G}_2$. Then there is an optimal path decomposition $\mathcal{P}$ of $G$ such that $\mathcal{P}|G_1$ has width at most $\mathbf{pw}(G_1) + t$.*

The following lemma shows that $\simeq_{\mathbf{pw},t}$ is a refinement of $\equiv_{\mathbf{pw},t}$.

**Lemma 6.7.** *Let $\widetilde{G}_1$ and $\widetilde{G}_2$ be two $t$-boundaried graphs. If $\widetilde{G}_1 \simeq_{\mathbf{pw},t} \widetilde{G}_2$, then $\widetilde{G}_1 \equiv_{\mathbf{pw},t} \widetilde{G}_2$.*

*Proof.* Let $\widetilde{G}_1$ and $\widetilde{G}_2$ be two $t$-boundaried graphs such that $\widetilde{G}_1 \simeq_{\mathbf{pw},t} \widetilde{G}_2$ and hence $\mathrm{FSCP}_{(\mathbf{pw}(G_1)+t)}(\widetilde{G}_1) = \mathrm{FSCP}_{(\mathbf{pw}(G_2)+t)}(\widetilde{G}_2)$. We show that $\mathbf{pw}(\widetilde{G}_1 \oplus \widetilde{H}) \leqslant \xi$ if and only if $\mathbf{pw}(\widetilde{G}_2 \oplus \widetilde{H}) \leqslant \xi$ for any $t$-boundaried graph $\widetilde{H}$ and any $\xi \in \mathbf{N_0}$. This implies $\widetilde{G}_1 \equiv_{\mathbf{pw},t} \widetilde{G}_2$ with $\Delta_{\mathbf{pw},t}(\widetilde{G}_1, \widetilde{G}_2) = 0$.

Let $\widetilde{H}$ and $\xi$ be such that $\mathbf{pw}(\widetilde{G}_1 \oplus \widetilde{H}) \leqslant \xi$. It follows from Corollary 6.6 that there is a path decomposition $\mathcal{P} = (P, \chi)$ of $\widetilde{G}_1 \oplus \widetilde{H}$ of width at most $\xi$ such that $\mathcal{P}|G_1$ is a path decomposition of $G_1$ of width at most $\mathbf{pw}(G_1) + t$. Hence, there is a characteristic in $\mathrm{FSCP}_{(\mathbf{pw}(G_1)+t)}(\widetilde{G}_1)$ corresponding to $\mathcal{P}|G_1$. Since $\mathrm{FSCP}_{(\mathbf{pw}(G_1)+t)}(\widetilde{G}_1) = \mathrm{FSCP}_{(\mathbf{pw}(G_2)+t)}(\widetilde{G}_2)$, we have that $\widetilde{G}_2$ has the same characteristic. It now follows from Proposition 6.2 that there is a path decomposition of $\widetilde{G}_2 \oplus \widetilde{H}$ that has the same width as $\mathcal{P}$ and hence $\mathbf{pw}(\widetilde{G}_2 \oplus \widetilde{H}) \leqslant$

$\xi$, as required. Because the reverse direction is analogous, this concludes the proof of the lemma. □

We are now ready to show the main result of this subsection, i.e., that the PATHWIDTH problem has FII on graphs of bounded pathwidth.

**Theorem 6.8.** *For $w \in \mathbf{N_0}$, let $\mathcal{PW}_w$ be a class of graphs that have pathwidth at most $w$. Then, the problem* PATHWIDTH *has FII in $\mathcal{PW}_w$.*

*Proof.* Because of Proposition 6.1 the number of equivalence classes of $\simeq_{\mathbf{pw},t}$ among graphs from $\mathcal{PW}_w$ is finite for every $t \in \mathbf{N_0}$. Furthermore, because of Lemma 6.7 it holds that $\simeq_{\mathbf{pw},t}$ is a refinement of $\equiv_{\mathbf{pw},t}$, which concludes the proof of the theorem. □

As bounded treedepth implies bounded pathwidth (see Proposition 2.12), using Theorems 4.1 and 4.9 we can conclude the following:

**Corollary 6.9.** PATHWIDTH *has a linear kernel in any graph class of bounded expansion, with the size $s$ of a modulator to constant treedepth as the parameter. Furthermore, under the same parameter $s$ it admits a kernel of near-linear size $\mathcal{O}(s^{1+\varepsilon})$ for every $\varepsilon > 0$ in any hereditary and nowhere dense graph class.*

### *6.2.* TREEWIDTH *has FII on graphs of small treewidth*

As the main ideas of the proof for treewidth are the same as for pathwidth (see the previous section), we present in detail only the first step, Lemma 6.10, which is different from former Lemma 6.5.

**Lemma 6.10.** *Let $\widetilde{G}_1$ and $\widetilde{G}_2$ be two $t$-boundaried graphs, $G = \widetilde{G}_1 \oplus \widetilde{G}_2$, and $\mathcal{T} = (T, \chi)$ be a tree decomposition of $G$. Then there is a tree decomposition $\mathcal{T}' = (T', \chi')$ of $G$ with the same width as $\mathcal{T}$ such that $\mathcal{T}'|G_1$ has width at most $\mathbf{tw}(G_1) + t$.*

*Proof.* If $\mathcal{T}|G_1$ has width at most $\mathbf{tw}(G_1)+t$, then $\mathcal{T}' := \mathcal{T}$ is the required tree decomposition of $G$. Hence, there is a bag $p \in V(T)$ such that $|\chi(p) \cap V(G_1)| > \mathbf{tw}(G_1)+t+1$. We call such a bag $p$ a *bad* bag of $\mathcal{T}$. The next claim shows that we can eliminate the bad bags of $\mathcal{T}$ one by one without introducing new bad bags. Hence, we obtain the desired tree decomposition $\mathcal{T}'$ from $\mathcal{T}$ by a repeated application of the following claim.

**Claim.** *There is a tree decomposition $\mathcal{T}'' = (T'', \chi'')$ of $G$ of the same width as $\mathcal{T}$ such that the set of bad bags of $\mathcal{T}''$ is a proper subset of the set of bad bags of $\mathcal{T}$. Moreover, the bag $p$ is no longer a bad bag of $\mathcal{T}''$.*

Let $\chi_{G_1}(p)$ be the set of vertices in $\chi(p) \cap V(G_1)$ and let $S$ be a minimum separator between $\chi_{G_1}(p)$ and $bd(G_1)$ in the graph $G$. Then, because $bd(G_1)$ is a separator between $\chi_{G_1}(p)$ and $bd(G_1)$ of cardinality at most $t$, we obtain that $|S| \leqslant t$. Let $W$ be the set of all vertices reachable from $\chi_{G_1}(p)$ in $G - S$, and let $\mathcal{T}_W = (T_W, \chi_W)$ be an optimal tree decomposition of $G[W]$. Then, because $W \subseteq V(G_1)$, it follows that the width $\mathcal{T}_W$ is at most the treewidth of $G_1$.

To obtain the desired tree decomposition $\mathcal{T}''$, where $p$ is not a bad bag anymore, we delete all vertices of $W$ from the bags of $\mathcal{T}$ and, instead, insert the tree decomposition $\mathcal{T}_W$ by connecting any node of $T_W$ via an edge to $p$ in $T$. However, to cover the edges between $S$ and $W$ in $G$ we also need to add $S$ to $p$ and every bag of $\mathcal{T}_W$. Because $\chi(p)$ does not necessarily contain all vertices of $S$, this could potentially violate the property P3 of a tree decomposition. To get around this we will add a vertex $s \in S$ to every bag $p' \in V(T)$ that is on a path between $p$ and any bag containing $s$ in $T$, i.e., we complete $\mathcal{T}''$ into a valid tree decomposition in a minimal way. This completes the construction of $\mathcal{T}''$ and it remains to argue that adding these vertices from $S$ does not increase the width of any bag in $\mathcal{T}$. Suppose it does, and let $p_2$ be a bag where we add more vertices than we remove. Let $S' \subseteq S$ be the set of added vertices and $R = \chi(p_2) \cap W$ the set of removed vertices. It follows that $|R| < |S'|$ and the bag $p_2$ separates in $T$ the set of bags containing a vertex from $S'$ from the bag $p$. Note that in $\mathcal{T}|G_1[W \cup S']$ we have $\chi_{G_1[W \cup S']}(p_2) = R$. Because of Proposition 2.6 applied to $\mathcal{T}|G_1[W \cup S']$, $R$ separates $\chi_{G_1[W \cup S']}(p)$ from $S'$ in $G_1[W \cup S']$.

We claim that $S'' = (S \setminus S') \cup R$ is a separator between $\chi_{G_1}(p)$ and $bd(G_1)$. Since $|S''| < |S|$, this would contradict the minimality of $S$. Let $\Pi$ be a path between $\chi_{G_1}(p)$ and $bd(G_1)$. Since $\chi_{G_1}(p) \subseteq W \cup S$, $\Pi$ has to intersect $S$ in order to reach $bd(G_1)$. Let $s$ be the first vertex of $\Pi$ which intersects $S$ (note that the subpath from $\chi_{G_1}(p)$ to $s$ of $\Pi$ lies entirely in $W$). Either $s \in S \setminus S'$ and therefore $s \in S''$, or $s \in S'$ and the subpath from $\chi_{G_1}(p)$ to $s$ of $\Pi$ lies entirely in $W \cup S'$, and therefore $\Pi$ has to intersect $R \subseteq S''$ in order to reach $s$. It follows that $S''$ is indeed a separator between $\chi_{G_1}(p)$ and $bd(G_1)$, completing the proof.

$\square$

**Corollary 6.11.** *Let $\widetilde{G}_1$ and $\widetilde{G}_2$ be two $t$-boundaried graphs and $G = \widetilde{G}_1 \oplus \widetilde{G}_2$. Then there is an optimal tree decomposition $\mathcal{T}$ of $G$ such that $\mathcal{T}|G_1$ has width at most* $\mathbf{tw}(G_1) + t$.

Employing a technical lemma analogous to Lemma 6.7, we obtain our main result of the subsection.

**Theorem 6.12.** *For $w \in \mathbf{N_0}$, let $\mathcal{TW}_w$ be a class of graphs that have treewidth at most $w$. Then, the problem* TREEWIDTH *has FII in $\mathcal{TW}_w$.*

*Proof.* The proof is analogous to the proof of Theorem 6.8. $\square$

Overall, we can conclude the section analogously to Corollary 6.9:

**Corollary 6.13.** TREEWIDTH *has a linear kernel in any graph class of bounded expansion, with the size $s$ of a modulator to constant treedepth as the parameter. Furthermore, under the same parameter $s$ it admits a kernel of near-linear size $\mathcal{O}(s^{1+\varepsilon})$ for every $\varepsilon > 0$ in any hereditary and nowhere dense graph class.*

## 7. Conclusions and Further Research

We have presented kernelization meta-results on graph classes of bounded expansion and on nowhere dense classes. More specifically, we have shown that all problems with FII on graphs of bounded treedepth admit linear problem kernels on graph classes of bounded expansion when parameterized by the size of a modulator to constant treedepth. For nowhere dense classes, we have shown that the kernels have almost-linear size.

The choice of our parameter (treedepth-modulator) is not arbitrary; as discussed in the introduction, e.g., a modulator to constant treewidth cannot yield linear kernels for certain natural problems that one would like to include in the framework. As argued before, this problem can be resolved *only* by choosing a parameter that generally increases when subdividing edges. Treedepth, which can be asymptotically characterized by absence of long paths as a subgraph, is thus a very natural choice for our purpose.

It remains an open question whether polynomial kernels (under a suitable weaker parameterization) exist for problems which are not invariant under edge subdivisions, such as Hamiltonian Cycle. Furthermore, our framework is general enough that it might apply to graph classes which are not part of the sparse graph hierarchy. A meta-kernel result for a dense graph class would be especially interesting. Recent work has shown that a linear kernel for classes of bounded expansion and an almost linear kernel for nowhere dense graph classes for Dominating Set exist when parameterized by the natural parameter [37]. This provides some hope that further problems admit such kernels since Dominating Set has acted as a catalyst for a flurry of results before (in fact, it was the problem that initiated the search for linear kernels on planar graphs).

Finally, it would be interesting to obtain a natural characterization of problems that have FII on graphs of bounded treedepth.

## References


[1] H. L. Bodlaender, F. V. Fomin, D. Lokshtanov, E. Penninkx, S. Saurabh, D. M. Thilikos, (Meta) Kernelization, in: Proc. of 50th FOCS, IEEE Computer Society, 2009, pp. 629–638.

[2] J. Guo, R. Niedermeier, Linear problem kernels for NP-hard problems on planar graphs, in: Proc. of 34th ICALP, Vol. 4596 of LNCS, Springer, 2007, pp. 375–386.

[3] J. Alber, M. R. Fellows, R. Niedermeier, Polynomial-time data reduction for Dominating Set, J. ACM 51 (2004) 363–384.

[4] F. V. Fomin, D. Lokshtanov, S. Saurabh, D. M. Thilikos, Bidimensionality and kernels, in: Proc. of 21st SODA, SIAM, 2010, pp. 503–510.

[5] E. J. Kim, A. Langer, C. Paul, F. Reidl, P. Rossmanith, I. Sau, S. Sikdar, Linear kernels and single-exponential algorithms via protrusion decompositions, ACM Trans. Algorithms 12 (2) (2016) 21.

[6] J. Nešetřil, P. Ossona de Mendez, Sparsity: Graphs, Structures, and Algorithms, Vol. 28 of Algorithms and Combinatorics, Springer, 2012.

[7] J. Nešetřil, P. Ossona de Mendez, Grad and classes with bounded expansion I. Decompositions, Eur. J. Comb. 29 (3) (2008) 760–776.

[8] H. Dell, D. van Melkebeek, Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses, in: Proc. of 42nd STOC, 2010, pp. 251–260.

[9] M. R. Fellows, B. M. P. Jansen, F. A. Rosamond, Towards fully multivariate algorithmics: Parameter ecology and the deconstruction of computational complexity, Eur. J. Comb. 34 (3) (2013) 541–566.

[10] H. L. Bodlaender, B. M. P. Jansen, S. Kratsch, Kernel bounds for path and cycle problems, in: Proc. of 6th IPEC, no. 7112 in LNCS, Springer, 2011, pp. 145–158.

[11] M. Cygan, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, S. Saurabh, On cutwidth parameterized by vertex cover, Algorithmica 68 (4) (2014) 940–953.

[12] M. R. Fellows, D. Lokshtanov, N. Misra, F. A. Rosamond, S. Saurabh, Graph layout problems parameterized by vertex cover, in: Proc. of 19th ISAAC, Vol. 5369 of LNCS, Springer, 2008, pp. 294–305.

[13] J. Fiala, P. A. Golovach, J. Kratochvíl, Parameterized complexity of coloring problems: Treewidth versus vertex cover, in: Proc. of 6th TAMC, no. 5532 in LNCS, Springer, 2009, pp. 221–230.

[14] M. Doucha, J. Kratochvíl, Cluster vertex deletion: a parameterization between vertex cover and clique-width, in: Proc. of 37th MFCS, no. 7464 in LNCS, Springer, 2012, pp. 348–359.

[15] R. Ganian, Twin-cover: Beyond vertex cover in parameterized algorithmics, in: Proc. of 6th IPEC, no. 7112 in LNCS, Springer, 2011, pp. 259–271.

[16] F. V. Fomin, B. M. Jansen, M. Pilipczuk, Preprocessing subgraph and minor problems: When does a small vertex cover help?, J. Comput. Syst. Sci. 80 (2) (2014) 468–495.

[17] M. Dom, D. Lokshtanov, S. Saurabh, Incompressibility through colors and IDs, in: Proc. of 36th ICALP, no. 5555 in LNCS, Springer, 2009, pp. 378–389.

[18] M. Cygan, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, S. Saurabh, On the hardness of losing width, Theory Comput. Syst. 54 (1) (2014) 73–82.

[19] H. L. Bodlaender, R. G. Downey, M. R. Fellows, D. Hermelin, On problems without polynomial kernels, J. Comput. Syst. Sci. 75 (8) (2009) 423–434.

[20] R. Diestel, Graph Theory, 4th Edition, Springer, Heidelberg, 2010.

[21] R. G. Downey, M. R. Fellows, Parameterized Complexity, Springer, 1999.

[22] J. Flum, M. Grohe, Parameterized Complexity Theory, Springer, 2006.

[23] J. Nešetřil, P. Ossona de Mendez, D. R. Wood, Characterisations and examples of graph classes with bounded expansion, Eur. J. Comb. 33 (3) (2012) 350–373.

[24] D. Wood, On the maximum number of cliques in a graph, Graphs Combin. 23 (2007) 337–352.

[25] F. V. Fomin, D. Lokshtanov, N. Misra, G. Philip, S. Saurabh, Hitting forbidden minors: Approximation and kernelization, in: Proc. of 28th STACS, Vol. 9 of LIPIcs, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2011, pp. 189–200.

[26] F. V. Fomin, D. Lokshtanov, N. Misra, S. Saurabh, Planar $\mathcal{F}$-deletion: Approximation and optimal FPT algorithms, in: Proc. of 53rd FOCS, IEEE Computer Society, 2012, pp. 470–479.

[27] B. Courcelle, The monadic second-order theory of graphs. I. Recognizable sets of finite graphs, Inform. Comput. 85 (1990) 12–75.

[28] J. Nešetřil, P. Ossona de Mendez, Linear time low tree-width partitions and algorithmic consequences, in: Proc. of 38th STOC, 2006, pp. 391–400.

[29] H. L. Bodlaender, T. Kloks, Better algorithms for the pathwidth and treewidth of graphs, in: Proc. of 18th ICALP, Vol. 510 of LNCS, Springer, 1991, pp. 544–555.

[30] H. L. Bodlaender, A linear time algorithm for finding tree-decompositions of small treewidth, SIAM J. Comput. 25 (1996) 1305–1317.

[31] J. Nešetřil, P. Ossona de Mendez, First order properties on nowhere dense structures, J. Symb. Logic 75 (3) (2010) 868–887.

[32] J. Nešetřil, P. Ossona de Mendez, On nowhere dense graphs, Eur. J. Comb. 32 (4) (2011) 600–617.

[33] B. de Fluiter, Algorithms for graphs of small treewidth, Ph.D. thesis, Utrecht University (1997).

[34] J. F. Geelen, A. M. H. Gerards, N. Robertson, G. Whittle, On the excluded minors for the matroids of branch-width $k$, J. Comb. Theory, Ser. B 88 (2) (2003) 261–265.

[35] I. V. Hicks, N. B. M. Jr., The branchwidth of graphs and their cycle matroids, J. Comb. Theory, Ser. B 97 (5) (2007) 681–692.

[36] H. L. Bodlaender, T. Kloks, Efficient and constructive algorithms for the pathwidth and treewidth of graphs, J. Algorithms 21 (2) (1996) 358–402.

[37] P. G. Drange, M. S. Dregi, F. V. Fomin, S. Kreutzer, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, F. Reidl, F. S. Villaamil, S. Saurabh, S. Siebertz, S. Sikdar, Kernelization and sparseness: the case of dominating set, in: Proc. of 33rd STACS, Vol. 47 of LIPIcs, 2016, pp. 31:1–31:14.

## 8. Appendix

In this appendix, we formally define some of the problems that we mention in this paper.

**LONGEST PATH**

| | |
|---|---|
| *Input:* | A graph $G$ and a positive integer $\ell$. |
| *Problem:* | Does $G$ contain a simple path of length at least $\ell$? |

**LONGEST CYCLE**

| | |
|---|---|
| *Input:* | A graph $G$ and a positive integer $\ell$. |
| *Problem:* | Does $G$ contain a simple cycle of length at least $\ell$? |

**EXACT $s, t$-PATH**

| | |
|---|---|
| *Input:* | A graph $G$, two special vertices $s, t \in V(G)$ and a positive integer $\ell$. |
| *Problem:* | Is there a simple path in $G$ from $s$ to $t$ of length exactly $\ell$? |

**EXACT CYCLE**

| | |
|---|---|
| *Input:* | A graph $G$ and a positive integer $\ell$. |
| *Problem:* | Is there a simple cycle in $G$ of length exactly $\ell$? |

**FEEDBACK VERTEX SET**

| | |
|---|---|
| *Input:* | A graph $G$ and a positive integer $\ell$. |
| *Problem:* | Is there a vertex set $S \subseteq V(G)$ with at most $\ell$ vertices such that $G - S$ is a forest? |

**TREEWIDTH**

| | |
|---|---|
| *Input:* | A graph $G$ and a positive integer $\ell$. |
| *Problem:* | Is the treewidth of $G$ at most $\ell$? |

**PATHWIDTH**

| | |
|---|---|
| *Input:* | A graph $G$ and a positive integer $\ell$. |
| *Problem:* | Is the pathwidth of $G$ at most $\ell$? |

TREEWIDTH-$t$ VERTEX DELETION

*Input:* A graph $G$ and a positive integer $\ell$.
*Problem:* Is there a vertex set $S \subseteq V(G)$ with at most $\ell$ vertices such that the treewidth of $G - S$ is at most $t$?

DOMINATING SET

*Input:* A graph $G = (V, E)$ and a positive integer $\ell$.
*Problem:* Is there a vertex set $S \subseteq V$ with at most $\ell$ vertices such that for all $u \in V \setminus S$ there exists $v \in S$ such that $uv \in E$?

If in addition, we require that $G[S]$ is a connected graph then the problem is called CONNECTED DOMINATING SET.

$r$-DOMINATING SET

*Input:* A graph $G = (V, E)$ and a positive integer $\ell$.
*Problem:* Is there a vertex set $S \subseteq V$ with at most $\ell$ vertices such that for all $u \in V \setminus S$ there exists $v \in S$ such that $d(u, v) \leqslant r$?

EFFICIENT DOMINATING SET

*Input:* A graph $G = (V, E)$ and a positive integer $\ell$.
*Problem:* Is there an independent set $S \subseteq V$ with at most $\ell$ vertices such that for every $u \in V \setminus S$ there exists exactly one $v \in S$ such that $uv \in E$?

EDGE DOMINATING SET

*Input:* A graph $G = (V, E)$ and a positive integer $\ell$.
*Problem:* Is there an edge set $S \subseteq E$ of size at most $\ell$ such that for every $e \in E \setminus S$ there exists $e' \in S$ such that $e$ and $e'$ share an endpoint?

INDUCED MATCHING

*Input:* A graph $G = (V, E)$ and a positive integer $\ell$.
*Problem:* Is there an edge set $S \subseteq E$ of size at least $\ell$ such that $S$ is a matching and for all $u, v \in V(S)$, if $uv \in E$ then $uv \in S$?

**CHORDAL VERTEX DELETION**

| | |
|---|---|
| *Input:* | A graph $G = (V, E)$ and a positive integer $\ell$. |
| *Problem:* | Is there a vertex set $S \subseteq V$ of size at most $\ell$ such that $G - S$ is chordal? |

**$\mathcal{F}$-MINOR-FREE DELETION**

| | |
|---|---|
| *Input:* | A graph $G = (V, E)$ and a positive integer $\ell$. |
| *Problem:* | Is there a vertex set $S \subseteq V$ of size at most $\ell$ such that $G - S$ does not contain any graph of the (finite) family $\mathcal{F}$ as a minor? |