The Price of Being Partial: Complexity of Partial Generalized Dominating Set on Bounded-Treewidth Graphs

Jakob Greilhuber
CISPA Helmholtz Center for Information Security, Saarbrücken, Germany
Saarland University, Saarbrücken, Germany

Dániel Marx
CISPA Helmholtz Center for Information Security, Saarbrücken, Germany

Abstract

For fixed sets σ, ρ of non-negative integers, the (σ, ρ)-domination framework introduced by Telle [*Nord. J. Comput.* 1994] captures many classical graph problems. For a graph G, a (σ, ρ)-set is a set S of vertices such that for every $v \in V(G)$, we have

(1) if $v \in S$, then $|N(v) \cap S| \in \sigma$, and

(2) if $v \notin S$, then $|N(v) \cap S| \in \rho$.

Algorithms and lower bounds for the decision, optimization, and counting versions of finding (σ, ρ)-sets on bounded-treewidth graphs were systematically studied [van Rooij et al., *ESA* 2009][Focke et al., *TALG* 2025]. We initiate the study of a natural partial variant (σ, ρ)-MinParDomSet of the problem, in which the constraints given by σ, ρ need not be fulfilled for all vertices, but we want to find a set of size at most k that maximizes the number of vertices that are satisfied in the sense that they satisfy (1) and (2) above.

Our goal is to understand whether (σ, ρ)-MinParDomSet can be solved in the same running time as the nonpartial version, or whether it is strictly harder. Formally, we consider nonempty finite or simple cofinite sets σ and ρ (simple cofinite sets are of the form $\mathbb{Z}_{\geq c}$), and we try to determine the smallest constant $c_{\sigma,\rho}$ such that there is a $c_{\sigma,\rho}^{\mathsf{tw}} \cdot n^{O(1)}$ time algorithm for the problem if a tree decomposition of width tw is given. We obtain matching upper and lower bounds on $c_{\sigma,\rho}$ for every such fixed σ and ρ under the Primal Pathwidth Strong Exponential Time Hypothesis, and establish whether the partial problem is harder than the nonpartial variant. For some sets σ and ρ, the more general (σ, ρ)-MinParDomSet has the same complexity as the nonpartial special case (e.g., for Dominating Set), while for other choices, the partial version is significantly harder (e.g., for Perfect Code).

2012 ACM Subject Classification Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases Generalized Dominating Set, Partial Domination, Treewidth, Primal Pathwidth Strong Exponential Time Hypothesis

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.97

Related Version *Full Version*: `https://arxiv.org/abs/2506.01645`

1 Introduction

Treewidth is one of the most-studied notions of capturing how tree-like a graph is, with significant algorithmic and graph-theoretic applications (see, for example, [28, Chapter 7]). Many graph problems can be solved in polynomial-time on trees, and similarly, many problems admit polynomial-time algorithms on graphs of bounded treewidth [9, 11, 12, 25, 28, 29, 43, 81, 85, 87, 89]. In fact, it is well-known that any graph problem expressible in (extensions of) monadic second-order logic can be solved in polynomial time on graphs of bounded treewidth [5, 25], which captures a large number of natural graph problems such as Hamiltonian Cycle, q-Coloring, and even optimization problems like Vertex Cover.

© Jakob Greilhuber and Dániel Marx;
licensed under Creative Commons License CC-BY 4.0

Showing that a problem can be solved in polynomial-time on graphs of bounded treewidth is often a standard application of dynamic programming techniques or algorithmic meta-theorems. However, improving the running time and obtaining optimal dependence on treewidth can be challenging. Let us illustrate this using the case of Dominating Set, one of the most fundamental algorithmic problems studied on graphs. On graphs of bounded treewidth, the initial algorithm by Telle and Proskurowski [85] solves the problem in time $9^{\mathsf{tw}} \cdot |I|^{O(1)}$, where tw is the width of a tree decomposition that is provided with the input I. Later, the running time was improved to $4^{\mathsf{tw}} \cdot |I|^{O(1)}$ [1, 2], and finally even $3^{\mathsf{tw}} \cdot |I|^{O(1)}$ [89] using the technique of Fast Subset Convolution [7]. Can we expect further improvements in the future? More generally, for any graph problem that has single-exponential dependency on the treewidth, what is the smallest c such that the problem can be solved in $c^{\mathsf{tw}} \cdot |I|^{O(1)}$?

Lokshtanov, Marx, and Saurabh [71] initiated a line of research aimed at addressing questions of this form, providing the first answers about the optimal dependence on treewidth for problems such as Dominating Set, Independent Set, and Max Cut. All of these problems can be solved in time $c^{\mathsf{tw}} \cdot |I|^{O(1)}$ for some problem-dependent constant c, but they show that an algorithm running in time $(c - \varepsilon)^{\mathsf{tw}} \cdot |I|^{O(1)}$ would refute the Strong Exponential Time Hypothesis (SETH) [19, 58]. Many further results showing lower bounds for problems on graphs of bounded treewidth under the SETH have appeared in the literature [4, 10, 14, 15, 26, 29–36, 39, 40, 49, 54–56, 62, 63, 67–69, 73, 74, 77–79]. These results usually confirm that the known algorithms for the studied problem are already essentially optimal, or reveal that there are cases where new nontrivial algorithmical insights are needed to match the lower bounds. In this line of work, it is a standard assumption that the tree decomposition is supplied as part of the input, as one wants to separate the complexity of computing such a decomposition from the complexity of solving the graph problem on such a decomposition.

However, even though the SETH is the standard complexity conjecture used in the field, it is not universally accepted to be true. Recently, Lampis [67] introduced the Primal Pathwidth Strong Exponential Time Hypothesis (PWSETH), partly to obtain more plausible lower bounds for the parameter pathwidth (and hence also for treewidth). The primal graph of a formula of k-SAT contains a vertex for each variable, and two vertices are connected by an edge if they appear together in some clause.

▶ **Hypothesis 1.1** (PWSETH [67, Conjecture 1.1]). *For any $\varepsilon > 0$, no algorithm can solve* 3-SAT *in time $(2 - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$, even when the input I contains a path decomposition of width* pw *of the primal graph of the input formula.*

Lampis [67] shows that the PWSETH is implied by the SETH, the Set Cover Conjecture [27], and the k-Orthogonal Vectors Assumption (see, e.g., [91]), making lower bounds obtained under it more plausible than those obtained under SETH alone. In the same work, Lampis proves tight PWSETH-based lower bounds for several problems, including Dominating Set parameterized by pathwidth, and shows that beating any of these lower bounds is equivalent to disproving the PWSETH. Even though the PWSETH is quite recent, there are already further publications presenting lower bounds under it [36, 55, 69]. All novel lower bounds we present in this paper are under the PWSETH.

Domination-like Problems. The (σ, ρ)-dominating set framework [84, 85] generalizes a wide range of classical graph problems by specifying desired neighborhood conditions through sets σ and ρ of integers. Formally, let σ and ρ be two fixed sets of non-negative integers. For a graph G, a (σ, ρ)-*set* of G is a set $S \subseteq V(G)$, such that

- for all $v \in S$ we have that $|N(v) \cap S| \in \sigma$, and
- for all $v \in V(G) \setminus S$ we have that $|N(v) \cap S| \in \rho$.

This flexible formulation captures many well-known problems. For example, a set S is a dominating set if and only if it is a $(\mathbb{Z}_{\geq 0}, \mathbb{Z}_{\geq 1})$-set, a p-dominating set if it is a $(\mathbb{Z}_{\geq 0}, \mathbb{Z}_{\geq p})$-set, a total dominating set if and only if it is a $(\mathbb{Z}_{\geq 1}, \mathbb{Z}_{\geq 1})$-set, and an independent set if and only if it is a $(\{0\}, \mathbb{Z}_{\geq 0})$-set. We refer to Telle [84] for a more extensive list of properties that can be expressed as (σ, ρ)-sets. Problems related to this domination concept have played a role in numerous publications since [13, 17, 21, 22, 37–39, 41, 42, 46, 47, 49, 52, 57, 61, 75, 86–89].

We are interested in the minimization problem (σ, ρ)-MinDomSet: given a graph G and integer k decide whether G has a (σ, ρ)-set of size at most k. Chappelle [22] shows that the problem can be solved with single-exponential dependency on the treewidth when σ, ρ are ultimately periodic sets, but does not explicitly give or optimize the base of the running time. Ultimately periodic sets capture, for example, finite and cofinite sets. As pointed out by Focke et al. [38], for general cofinite sets, the complexity of (σ, ρ)-MinDomSet on bounded-treewidth graphs is somewhat mysterious: it depends on the power of representative sets, which is not understood very well (see also [50, 74]). Therefore, we restrict our attention to finite or *simple cofinite* sets (sets of the form $\mathbb{Z}_{\geq c}$ for some integer c) in this work. Most classical graph problems that can be expressed using the (σ, ρ)-domination framework have sets which are finite or simple cofinite. In fact, all problems given by Telle [84, Table I] as examples for classical problems that the framework can describe have such sets.

To explain the concrete running times of the algorithms, we define some constants which correspond to the largest state a dynamic programming algorithm requires for each set.

▶ **Definition 1.2** (Constants s_σ and s_ρ). *Define*

$$s_\sigma = \begin{cases} \max \sigma & \text{if } \sigma \text{ is finite,} \\ \min \sigma & \text{if } \sigma \text{ is simple cofinite,} \end{cases} \quad \text{and } s_\rho = \begin{cases} \max \rho & \text{if } \rho \text{ is finite,} \\ \min \rho & \text{if } \rho \text{ is simple cofinite.} \end{cases}$$

Let $s_{\sigma,\rho} = \max(s_\rho, s_\sigma)$.

When σ and ρ are both finite or simple cofinite, van Rooij [88] shows that (σ, ρ)-MinDomSet can be solved in time $(s_\sigma + s_\rho + 2)^{\mathsf{tw}} \cdot |I|^{O(1)}$, when a tree decomposition of width tw is provided as part of the input I. This running time is seemingly optimal, as the base corresponds exactly to the number of states in the natural dynamic programming approach to the problem. However, Focke et al. [38] show that if the sets σ and ρ satisfy some additional combinatorial properties, then improved algorithms are possible.

The pair (σ, ρ) is *m-structured* if σ is a subset of some residue class modulo m, and ρ is a subset of some residue class modulo m. While every pair (σ, ρ) is 1-structured by definition, Focke et al. [38] show that improvements over the previously best known algorithms are possible when (σ, ρ) is m-structured for some $m \geq 2$.

▶ **Theorem 1.3** (Due to results of [38, 88]). *Let σ, ρ be nonempty finite or simple cofinite sets. When a tree decomposition of width tw of the input graph is provided as part of the input I, the problem (σ, ρ)-MinDomSet can be solved in time*

- $(s_\sigma + s_\rho + 2)^{\mathsf{tw}} \cdot |I|^{O(1)}$ *when (σ, ρ) is not m-structured for any $m \geq 2$,*
- $(\max(s_\sigma, s_\rho) + 1)^{\mathsf{tw}} \cdot |I|^{O(1)}$ *when (σ, ρ) is m-structured for some $m \geq 3$, or (σ, ρ) is 2-structured and it is not the case that $s_\sigma = s_\rho$ is even,*
- $(s_\sigma + 2)^{\mathsf{tw}} \cdot |I|^{O(1)}$ *otherwise, that is when (σ, ρ) is 2-structured but not m-structured for any $m \geq 3$, and $s_\sigma = s_\rho$ is even.*

Are the upper bounds in Theorem 1.3 tight for every σ and ρ? Focke et al. [39] show that these algorithms are tight under the SETH when σ, ρ are finite sets (and $0 \notin \rho$). Later, the same result was proven under the PWSETH in the thesis of Schepper [82]. We complete the picture by extending the lower bound to every finite and simple cofinite set. While this result establishes tight lower bounds for some fundamental problems where no such bounds were known (e.g., p-Dominating Set, Independent Dominating Set – see also [77] for other open questions that we answer), it is secondary to our main contribution about the partial problem variants. However, we need to have these results for (σ, ρ)-MinDomSet at hand in order to be able to contrast them with our tight bounds for the partial problems.

▶ **Main Theorem 1.** *Let σ, ρ be nonempty finite or simple cofinite sets with $0 \notin \rho$. Then, unless the PWSETH is false, for any $\varepsilon > 0$, and even when the input I contains a path decomposition of width* pw *of the input graph, there is no algorithm that can solve the problem (σ, ρ)-MinDomSet in time*

- $(s_\sigma + s_\rho + 2 - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$ *when (σ, ρ) is not m-structured for any $m \geq 2$,*
- $(\max(s_\sigma, s_\rho) + 1 - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$ *when (σ, ρ) is m-structured for some $m \geq 3$, or (σ, ρ) is 2-structured and it is not the case that $s_\sigma = s_\rho$ is even,*
- $(s_\sigma + 2 - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$ *otherwise, that is when (σ, ρ) is 2-structured but not m-structured for any $m \geq 3$, and $s_\sigma = s_\rho$ is even.*

Partial Optimization Problems. For a minimization problem where the goal is to find a solution of at most a certain size that satisfies every constraint, we can consider the natural relaxation where the goal is no longer to satisfy every constraint, but rather a specified portion of it. This relaxation introduces new algorithmic and combinatorial challenges. Various partial optimization problems have been studied both from the perspective of exact and parameterized algorithms, as well as approximation and heuristics [6, 16, 23, 24, 44, 53, 59, 76, 83, 90]. For example, for Dominating Set the partial problem means looking for a set of size at most k that dominates at least ℓ vertices, or for Vertex Cover finding a set of vertices of size at most k that covers at least ℓ edges. These problems known as Partial Dominating Set and Partial Vertex Cover have received considerable attention [3, 8, 18, 20, 45, 48, 51, 60, 64, 65, 72].

Our Contribution. In this paper, we initiate the study of the natural partial variant of the (σ, ρ)-MinDomSet problem on graphs of bounded treewidth. We begin by defining the appropriate notion of domination, which we refer to as satisfying vertices.

▶ **Definition 1.4.** *Let σ and ρ be two sets of non-negative integers. Let G be a graph and $S \subseteq V(G)$. A vertex $v \in V(G)$ is* satisfied *(by S relative to (σ, ρ)) if:*

1. *when $v \in S$, then $|N(v) \cap S| \in \sigma$,*
2. *when $v \notin S$, then $|N(v) \cap S| \in \rho$.*

A vertex $v \in V(G)$ that is not satisfied is called violated *(by S relative to (σ, ρ)).*

Next, we formally introduce the partial problem variant (σ, ρ)-MinParDomSet.

(σ, ρ)-MinParDomSet

Input:	Graph G, integers k and ℓ
Question:	Is there a set $S \subseteq V(G)$ such that S has size at most k and at most ℓ vertices are violated by S relative to (σ, ρ)?

We study (σ, ρ)-MinParDomSet on graphs of bounded treewidth for sets σ, ρ which are nonempty finite sets, or simple cofinite sets. Observe that the problem generalizes (σ, ρ)-MinDomSet, in which one looks for a set that satisfies all vertices. So, (σ, ρ)-MinParDomSet is at least as hard as (σ, ρ)-MinDomSet. The main question we want to answer is whether this hardness is strict.

*For a given σ and ρ, is the partial problem (σ,ρ)-*MinParDomSet *strictly harder on bounded-treewidth graphs than the corresponding (σ,ρ)-*MinDomSet *problem?*

We answer this question by a complete characterization of the complexity of (σ,ρ)-MinParDomSet for every nonempty σ and ρ that are finite or simple cofinite: we provide an algorithm and show that it is tight under the PWSETH.

Moving from (σ,ρ)-MinDomSet to (σ,ρ)-MinParDomSet does not always result in a harder problem. Let us illustrate this for Partial Dominating Set. We can observe that, similarly to Dominating Set, a dynamic programming algorithm on a tree decomposition needs to consider three states for a vertex v: (1) v is unselected with no selected neighbor so far, (2) v is unselected with a selected neighbor, and (3) v is selected (see also [60]). Furthermore, Fast Subset Convolution can be performed similarly to Dominating Set [89], resulting in running time $3^{\mathsf{tw}} \cdot |I|^{O(1)}$ [70] (earlier, a $4^{\mathsf{tw}} \cdot |I|^{O(1)}$ algorithm was presented by Roayaei and Razzazi [80]). The observation about the number of states can be generalized to the p-Dominating Set problem, which is (σ,ρ)-MinDomSet with $\sigma = \mathbb{Z}_{\geq 0}$ and $\rho = \mathbb{Z}_{\geq p}$. Both for the partial and nonpartial variant, we need to keep track of $p+2$ states: vertex v is selected, vertex v is unselected with $i = 0,\dots,p-1$ neighbors, or vertex v is unselected with at least p neighbors. With these states it is possible to solve the problem in time $(p+2)^{\mathsf{tw}} \cdot |I|^{O(1)}$ using fast convolution algorithms. Note that the important observation that the number of states for p-Dominating Set is the same in the partial and nonpartial versions was already made by Liu and Lu [70], even though the running time they obtain is worse than $(p+2)^{\mathsf{tw}} \cdot |I|^{O(1)}$ due to inefficient handling of join nodes.

The situation changes when at least one of σ or ρ is a finite set. If, say, σ is finite, then in the nonpartial problem a selected vertex v can have $s_\sigma + 1 = \max\sigma + 1$ possible states: vertex v can have $0,\dots,\max\sigma$ selected neighbors. On the other hand, in the partial problem, we need an additional "overflow" state to express the case when the vertex has already at least $\max\sigma + 1$ selected neighbors, making it violated (and no amount of further selected neighbors can make it satisfied). Thus, it seems that whenever σ or ρ is finite, we need one extra state and have a corresponding increase in the running time. We confirm this intuition.

Let us now formally state our main results. To describe the running time of our algorithm, we define some constants based on σ and ρ, which again have a direct correspondence to the largest states required for σ,ρ in the dynamic programming algorithm. The superscript p refers to "partial", in order to distinguish the constants from s_σ and s_ρ.

▶ **Definition 1.5** (Constants s^{p}_σ and s^{p}_ρ). *Define*

$$s^{\mathsf{p}}_\rho = \begin{cases} \max\rho + 1 & \text{if } \rho \text{ is finite,} \\ \min\rho & \text{if } \rho \text{ is simple cofinite,} \end{cases} \quad \text{and } s^{\mathsf{p}}_\sigma = \begin{cases} \max\sigma + 1 & \text{if } \sigma \text{ is finite,} \\ \min\sigma & \text{if } \sigma \text{ is simple cofinite.} \end{cases}$$

Let $s^{\mathsf{p}}_{\sigma,\rho} = \max(s^{\mathsf{p}}_\rho, s^{\mathsf{p}}_\sigma)$.

Our first main result about the partial problem is an algorithm for (σ,ρ)-MinParDomSet.

▶ **Main Theorem 2.** *Let σ,ρ be fixed nonempty finite or simple cofinite sets. Then, there is an algorithm that can, when given a graph G and a tree decomposition of G of width* tw *as input I, decide, for every $k,\ell \in [0,|V(G)|]$ at the same time, whether there exists a set $S \subseteq V(G)$ of size exactly k violating exactly ℓ vertices relative to (σ,ρ), in time $(s^{\mathsf{p}}_\sigma + s^{\mathsf{p}}_\rho + 2)^{\mathsf{tw}} \cdot |I|^{O(1)}$.*

We then show that under the PWSETH these algorithms cannot be improved, unless $0 \in \rho$, in which case the empty set is a trivial optimal solution to the problem.

Table 1 Comparison of the tight running times for the nonpartial and partial minimization variants for several classical domination-like problems. Polynomial factors in the input size are hidden. The bounds for the nonpartial variant stem from Theorem 1.3 and Main Theorem 1, the bounds for the partial variant from Main Theorems 2 and 3. Most problems of this table are taken from [84, Table I].

Problem Name	σ	ρ	Nonpartial	Partial
Dominating Set	$\mathbb{Z}_{\geq 0}$	$\mathbb{Z}_{\geq 1}$	3^{tw}	3^{tw}
p-Dominating Set ($p \geq 1$)	$\mathbb{Z}_{\geq 0}$	$\mathbb{Z}_{\geq p}$	$(p+2)^{\mathsf{tw}}$	$(p+2)^{\mathsf{tw}}$
Perfect Code	$\{0\}$	$\{1\}$	2^{tw}	5^{tw}
Perfect Dominating Set	$\mathbb{Z}_{\geq 0}$	$\{1\}$	3^{tw}	4^{tw}
Total Dominating Set	$\mathbb{Z}_{\geq 1}$	$\mathbb{Z}_{\geq 1}$	4^{tw}	4^{tw}
Total Perfect Dominating Set	$\{1\}$	$\{1\}$	2^{tw}	6^{tw}
Weakly Perfect Dominating Set	$\{0,1\}$	$\{1\}$	4^{tw}	6^{tw}
Independent Dominating Set	$\{0\}$	$\mathbb{Z}_{\geq 1}$	3^{tw}	4^{tw}
Dominating p-Regular Subgraph ($p \geq 0$)	$\{p\}$	$\mathbb{Z}_{\geq 1}$	$(p+3)^{\mathsf{tw}}$	$(p+4)^{\mathsf{tw}}$

▶ **Main Theorem 3.** *Let σ, ρ be nonempty finite or simple cofinite sets with $0 \notin \rho$. Then, unless the PWSETH is false, there is no algorithm that can solve the problem (σ,ρ)-*MinParDomSet *in time $(s^{\mathsf{p}}_\sigma + s^{\mathsf{p}}_\rho + 2 - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$ for any $\varepsilon > 0$, even when a path decomposition of the input graph of width* pw *is provided as part of the input I.*

When both σ and ρ are simple cofinite, for example for the p-Dominating Set problem, we see that the algorithms for the partial and nonpartial variant have the same running time. On the other hand, comparing Theorem 1.3 and Main Theorem 3, we can see that when at least one of σ, ρ is finite, then the running time for the partial version is strictly higher than the running time for the nonpartial version. We provide a comparison of the obtained running times for a variety of classical problems in Table 1.

A particularly striking example is the case of $\sigma = \{1\}, \rho = \{1\}$, where the problem (σ,ρ)-MinDomSet can be solved in time $2^{\mathsf{tw}} \cdot |I|^{O(1)}$ [38], but (σ,ρ)-MinParDomSet requires time $6^{\mathsf{tw}} \cdot |I|^{O(1)}$. We identify two main reasons for these differing running times.

- **Overflow states are needed for finite sets.** As discussed above, if a set $\tau \in \{\sigma, \rho\}$ is simple cofinite, there is no difference in the state set needed for the partial and nonpartial variants (we have $s_\tau = s^{\mathsf{p}}_\tau$). On the other hand, if τ is finite, then the size of the state set for the partial variant is larger than for the nonpartial variant (we have $s^{\mathsf{p}}_\tau > s_\tau$). This indicates that we have to pay an additional price for each finite set.
- **Structured sets are irrelevant.** For m-structured sets (σ, ρ) (with $m \geq 2$) the algorithm by Focke et al. [38] is based on a parity argument that can significantly reduce the number of subproblems that need to be considered. However, if some vertices of the graph are violated in the solution, then this combinatorial argument no longer holds. Hence, the running time difference is especially large for m-structured sets (σ, ρ).

Main Theorem 3 is proven in technical reduction chains, and these proofs represent the main bulk of the work done for this paper. In Section 2, we present an overview of the used techniques.

2 Technical Overview

We will now give an overview of the technical ideas used to obtain our results.

2.1 The Algorithm

For our algorithmic result, we utilize standard dynamic programming procedures, similar to those given in [38, 49] for other variants of (σ, ρ)-dominating set. Using typical notation for these types of problems, we introduce certain state sets.

▶ **Definition 2.1** (State Sets for the Partial Problem). *Let σ, ρ be nonempty finite or simple cofinite sets. We define the set of ρ-states for the partial problem as $\mathbb{R}_{\mathsf{p}} = \{\rho_0, \ldots, \rho_{s^{\mathsf{p}}_\rho}\}$, and the set of σ-states for the partial problem as $\mathbb{S}_{\mathsf{p}} = \{\sigma_0, \ldots, \sigma_{s^{\mathsf{p}}_\sigma}\}$. We denote the set of all states for the partial problem as $\mathbb{A}_{\mathsf{p}} = \mathbb{S}_{\mathsf{p}} \cup \mathbb{R}_{\mathsf{p}}$.*

When τ is a finite set, the state $\tau_{s^{\mathsf{p}}_\tau}$ is called an overflow-state.

In a partial solution, each vertex is assigned one of these states. If a vertex is selected it has a σ-state, if it is unselected it has a ρ-state. The subscript of the state describes the number of selected neighbors the vertex has. Observe that we have $|\mathbb{A}_{\mathsf{p}}| = s^{\mathsf{p}}_\sigma + s^{\mathsf{p}}_\rho + 2$, which forms the base of the running time in Main Theorem 2.

We proceed by dynamic programming on a nice tree decomposition, for a node t, X_t is the bag of t, and V_t is the set of vertices introduced in and below node t. For a node t and a vector $\vec{u} \in \mathbb{A}_{\mathsf{p}}^{X_t}$, we say that $\vec{u}$ has a matching partial solution of size k violating ℓ vertices if there is a set $S \subseteq V_t$ such that:

- $|S \setminus X_t| = k$, and
- S violates exactly ℓ vertices of $V_t \setminus X_t$, and
- for any $v \in X_t$, $\vec{u}[v]$ is a σ-state if and only if $v \in S$, and when $\vec{u}[v] = \tau_c$, we have that $c = \min(x, s^{\mathsf{p}}_\tau)$, where x is the number of neighbors that vertex v has in $S \setminus X_t$.

For each node t, and each possible solution size and violation number, we have a table entry $T_t[k, \ell]$, whose task is to store the set of all vectors of $\mathbb{A}_{\mathsf{p}}^{X_t}$ that have a matching partial solution for t of size k violating ℓ vertices. Using fast convolution algorithms by van Rooij [88], one can compute these table entries in time $|\mathbb{A}_{\mathsf{p}}|^{\mathsf{tw}} \cdot |I|^{O(1)}$, which yields Main Theorem 2.

2.2 The Lower Bound

Next we describe the main framework of the lower bound, which is the far more technical part of our contribution.

2.2.1 The Three-Step Framework

On a very high level, our strategy follows a line of work in the domain of tight lower bound for dynamic programming algorithms [26, 39, 49, 73, 74]. In these results, a certain design pattern has emerged, which we also follow. Typically, the proof is by reduction from k-SAT, or an appropriate CSP variant. If our goal is to rule out the existence of $(c - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$ time algorithms, then it is convenient to start the reduction from a CSP problem with domain size c, for which Lampis [66] provides a lower bound of the correct form under SETH.

In these reductions, the pathwidth pw of the graph of the output instance needs to be roughly the number n of variables of the input CSP instance. Typically, the construction starts by introducing an $n \times m$ grid of *state-vertices,* where m can be arbitrarily large. Any two consecutive columns of state vertices are connected by low-pathwidth gadgets, ensuring that the constructed graph has pathwidth roughly n.

Given a solution of the constructed instance a state is assigned to each state vertex v based on the role that v plays in the solution. For the problem (σ, ρ)-MinParDomSet the state is determined by whether v is in the solution and by the number of selected neighbors it has in the gadgets connecting the column of v to the *previous* column. Thus, the states of the n vertices in a column can be interpreted as a variable assignment of the source CSP instance with n variables and domain size $|\mathbb{A}_{\mathsf{p}}|$. To obtain a correct reduction, the gadgets connecting the columns need to enforce two properties:

1. Each column must express exactly the same assignment.
2. The assignment satisfies the input instance.

To ensure the first property, the gadgets between columns i and $i+1$ need to "copy" the states of the vertices in column i to column $i+1$. Furthermore, given that the number m of columns can be made arbitrary large without changing the pathwidth, it is convenient to require that the gadgets between columns i and $i+1$ ensure that the i-th constraint of the source CSP instance is satisfied.

For every fixed σ and ρ, we need to be able to construct gadgets that are responsible for the tasks described above. However, when designing gadgets of different types, at some point it is easier to observe and prove that we are able to construct *every* gadget (in a well-defined sense). Formally, we introduce a new, more general problem where the input contains a set of "relations": additional constraints on certain subsets of vertices. By exploiting the additional power of these relations the lower bound is proved for this more general problem. Then, the relations are systematically replaced by gadgets.

Instead of having one large reduction, we can use a framework that splits the reduction into three steps, each having a clear individual goal. Let $\mathcal{P}$ be the problem one wants to show a lower bound for.

1. A new problem $\mathcal{P}'$ is defined. Problem $\mathcal{P}'$ is a variant of $\mathcal{P}$, in which the input instance contains a list of *relations* that introduce additional constraints a solution must fulfill. For example, in the (σ, ρ)-domination problem, a relation can constrain a subset X of vertices, restricting which combinations of vertices of X can be selected in a solution. We refer to problem $\mathcal{P}'$ as the *intermediate problem* or the *problem with relations*. A lower bound is shown for problem $\mathcal{P}'$ using a grid-like construction.
2. The relations of the problem $\mathcal{P}'$ are replaced by a set of much simpler relations.
3. Finally, the simple relations are replaced by gadgets that simulate their behavior in a reduction to problem $\mathcal{P}$.

This framework was used by Curticapean and Marx [26] to show a lower bound for counting the number of perfect matchings of a graph. Later, the framework was used for edge-selection problems [73, 74]. Finally, Focke et al. [39] and Greilhuber et al. [49] utilize this approach for (σ, ρ)-domination problems.

2.2.2 Challenges

We are using the same framework for our problem. We can profit from the fact that the first two steps are already somewhat well-understood for (σ, ρ)-domination problems. However, we still face multiple challenges which make the application of the framework far from straightforward, one of them requires a novel spin on step 1.

Challenges in step 1. Previous applications of the framework for (σ, ρ)-domination problems use the same definition for the intermediate problem. In this definition, a relation is a pair $(\mathsf{scp}(R), \mathsf{acc}(R))$, where $\mathsf{scp}(R) \subseteq V(G)$ is the scope of the relation, and $\mathsf{acc}(R) \subseteq 2^{\mathsf{scp}(R)}$ the set of accepted selections. A solution S to the problem with relations must not only be a (σ, ρ)-set, but also for each relation R it must be the case that $S \cap \mathsf{scp}(R) \in \mathsf{acc}(R)$. A path decomposition of an instance of the intermediate problem is a path decomposition of the graph such that additionally for each relation R there exists a bag containing $\mathsf{scp}(R)$. Moreover, the size of the scope of each relation that is used in the reductions should be bounded by a constant. These two properties ensure that steps 2 and 3 do not increase the pathwidth too much.

The first major challenge we face is about the difference between the partial and nonpartial problem. The constructions used for step 1 in the literature [39, 49] are for the nonpartial problem, and thus a solution cannot violate any vertices. When dealing with the partial problem, we have an increase in the running times due to the fact that overflow-states are required for finite sets. We need to ensure that some of our state vertices can have overflow-states in a solution, otherwise we would not get a higher base in the lower bound than for the nonpartial problem. Hence, we must carefully create a construction in which some state vertices are violated and correctly propagate their state. In particular, a priori it is not clear why a violated state vertex should have its state propagate across the construction.

The second major problem is that the intermediate problem defined above is too powerful when ρ is a simple cofinite set. It turns out that when ρ is simple cofinite, there are relations which cannot be simulated by gadgets in a straightforward way. Therefore, we need to work with a more restricted problem where $\mathsf{acc}(R)$ is assumed to be superset-closed for each relation R. We can prove the required lower bound even for this significantly less expressive problem, and the superset-closed relations can be simulated with appropriate gadgets.

Challenges in step 2. A $\mathtt{HW}_{=1}$-relation (hamming weight one relation) accepts a selection from its scope if and only if it has size exactly one. Focke et al. [39] showed that (as long as $\rho \neq \{0\}$) there is a reduction from (σ, ρ)-MinDomSet with arbitrary relations with bounded arity to the problem using only $\mathtt{HW}_{=1}$-relations of bounded arity. While the reduction also works for the partial problem (σ, ρ)-MinParDomSet, it would not be helpful when ρ is simple cofinite, since we cannot replace the $\mathtt{HW}_{=1}$ relations by gadgets. We instead show that superset-closed relations can be replaced by $\mathtt{HW}_{\geq 1}$-relations, where a $\mathtt{HW}_{\geq 1}$ relation accepts a selection from its scope if and only if it is nonempty.

Challenges in step 3. In step 3, we need to replace the simple relations ($\mathtt{HW}_{=1}$ or $\mathtt{HW}_{\geq 1}$) we obtain after step 2 with gadgets. The construction of these gadgets is very specific to the (σ, ρ)-MinParDomSet problem and highly depends on the properties of the sets σ and ρ.

The gadgets given in the literature [39, 49] cannot directly be used here, since they are for the nonpartial variant. Hence, the first major challenge we face in step 3 is that we must create gadgets that can handle violations. Thus, the first step is to define gadgets such that if the behavior is not as desired, then the gadget has to pay a "penalty" either in terms of violated vertices or larger solution size. The correct form of the definition is not obvious: in fact, we need to use two greatly different definitions for the two cases when ρ is finite or simple cofinite ("robust" and "fragile" realizations, see Definitions 2.4 and 2.10).

2.2.3 The Reduction Chains

Let us now present the reduction chain of our lower bound in more detail. We will present a "plan of attack" which illustrates which definitions and lemmas we use to handle steps 1-3 of the framework. The reduction chain is different depending on whether ρ is finite or simple cofinite. We focus on the partial problem variant and omit details about Main Theorem 1 in this overview. The two reductions in step 1 below start from a CSP variant for which Lampis [67, Theorem 3.2] proves the required lower bound under the PWSETH.

Set ρ is finite. We begin by covering the case when ρ is finite, as our reduction chain is closer to the known approaches in this scenario.

Step 1. For our intermediate problem, we use the natural partial variant of the problem with relations. A relation R is a pair $(\mathsf{scp}(R), \mathsf{acc}(R))$ where $\mathsf{scp}(R)$ is a vertex set and $\mathsf{acc}(R) \subseteq 2^{\mathsf{scp}(R)}$.

(σ, ρ)-ParDomSet$_{\mathcal{R}}$

Input: Graph G, integer ℓ, set of relations $\mathcal{R}$

Question: Is there a set $S \subseteq V(G)$ such that at most ℓ vertices of $V(G)$ are violated by S relative to (σ, ρ), and for each $R \in \mathcal{R}$ it holds that $S \cap \mathsf{scp}(R) \in \mathsf{acc}(R)$?

An instance of (σ, ρ)-ParDomSet$_{\mathcal{R}}$ has arity d (or at most d), when $|\mathsf{scp}(R)| \leq d$ for each $R \in \mathcal{R}$. A path decomposition of an instance of (σ, ρ)-ParDomSet$_{\mathcal{R}}$ is a path decomposition of the input graph G such that additionally for each relation R there is a bag that contains $\mathsf{scp}(R)$.

Since (σ, ρ)-ParDomSet$_{\mathcal{R}}$ is a partial variant of the intermediate problem used in the literature [39], we can propagate the states of *satisfied* state vertices in essentially the same way in our grid-like construction. However, changes are required since we also need to be able to propagate the states of *violated* state vertices across the construction. This can be achieved by only allowing certain vectors of states to be used. Formally, we show the following intermediate lower bound for (σ, ρ)-ParDomSet$_{\mathcal{R}}$.

▶ **Lemma 2.2.** *Let σ, ρ be nonempty finite or simple cofinite sets. For any $\varepsilon > 0$, there is a constant d depending only on ε, σ, and ρ, such that if there is an algorithm that solves the problem (σ, ρ)-ParDomSet$_{\mathcal{R}}$ on instances of arity at most d, where a path decomposition of width pw of the instance is provided as part of the input I, in time $(|\mathbb{A}_{\mathsf{p}}| - \varepsilon)^{\mathsf{pw}} \cdot |I|^{O(1)}$, then the PWSETH is false.*

To prove the lemma above, we provide a suitable reduction for each fixed ε.

Step 2. Next, we simplify the used relations. Using a reduction by Focke et al. [39, Corollary 8.8], which also works for the partial problem variant, we get the following result.

▶ **Lemma 2.3.** *Let σ, ρ be nonempty sets with $\rho \neq \{0\}$. There is a pathwidth-preserving reduction from arbitrary instances (σ, ρ)-ParDomSet$_{\mathcal{R}}$ with arity at most d to instances (σ, ρ)-ParDomSet$_{\mathcal{R}}$ with arity at most $2^d + 1$ that only use $\mathtt{HW}_{=1}$-relations.*

Step 3. All that remains is to find gadgets that can replace the $\mathtt{HW}_{=1}$-relation. For the nonpartial problem and finite σ, ρ, Focke et al. [39] show that there are graphs of constant size which do the job. To replace a relation R on the vertices $\mathsf{scp}(R)$, they add a small gadget to the graph that is connected to $\mathsf{scp}(R)$. This gadget does not provide any further selected vertices to $\mathsf{scp}(R)$ in any (σ, ρ)-set, but it ensures that exactly one vertex of $\mathsf{scp}(R)$ must be selected. Unfortunately, we cannot directly use the gadgets given in the literature, as they do not function correctly when vertices can be violated.

To overcome this, we build gadgets that are robust against violations. Intuitively, this can be achieved by ensuring that if the gadget does not behave correctly, then there is some additional price to pay.[1] We formally define the gadgets we utilize next.

[1] Note that in principle the idea of using gadgets that simulate relations with some notion of penalty for undesired behaviors is not new. For instance, Marx et al. [73] use this approach for a class of edge selection problems.

▶ **Definition 2.4** (Robust Realization). *Let σ, ρ be sets of non-negative integers. A pair (G, U), where G is a graph and $U \subseteq V(G)$ realizes the $\texttt{HW}_{=1}$-relation of arity $|U|$ with penalty δ, cost tradeoff β, and cost γ if the following properties all hold:*

1. *For any $u \in U$, there exists a set $S_u \subseteq V(G)$ such that $S_u \cap U = \{u\}$, $S_u \cap N(U) = \emptyset$, $|S_u \setminus U| = \gamma$, and S_u violates no vertex of $V(G) \setminus U$.*
2. *For any set $S \subseteq V(G)$ that violates at most ℓ vertices of $V(G) \setminus U$, it holds that $S \setminus U$ has size at least $\gamma - \ell \cdot \beta$.*
3. *Any set $S \subseteq V(G)$ such that $|S \cap U| \neq 1$ violates at least one vertex of $V(G) \setminus U$ or fulfills $|S \setminus U| > \gamma$.*
4. *Any set $S \subseteq V(G)$ such that $S \cap N(U) \neq \emptyset$ violates more than δ vertices of $V(G) \setminus U$ or fulfills $|S \setminus U| > \gamma + \delta$.*
5. *U is an independent set of G.*

When replacing a $\texttt{HW}_{=1}$-relation, the relation scope is identified with the set U of a fresh copy of the robust realization gadget. So, the goal of such a gadget is ensuring that exactly one vertex of U is selected, and that no vertex of $N(U)$ is selected. Let us elaborate on the definition. First, there is a small solution (selecting at most γ vertices outside U), whenever exactly one vertex u of U is selected. The solution also has the property that it selects no vertex of $N(U)$. This guarantees that whenever exactly one vertex of U is selected we can extend the selection of U to this solution of the robust realization gadget without changing the number of selected neighbors of vertices of U. On the other hand, if a solution is not of this form, then we have to pay for it either in terms of larger solution size or number of violated vertices. Specifically, the tradeoff value β of the second property guarantees that decreasing the solution size is possible only at the cost of some number of violated vertices. The third property says that if not exactly one vertex of U is selected, then either there is a violation or the solution size strictly increases. The fourth property states that selecting a vertex in the neighborhood of U is even worse: it creates more than δ violations or increases solution size by more than δ.

We show that these types of gadgets exist for arbitrarily large penalties δ. Our constructions can be seen as extensions of those given in the literature [39] that take solution sizes into account and can also handle violations.

▶ **Lemma 2.5.** *Let σ be a nonempty finite or simple cofinite set, and ρ be a finite set with $0 \notin \rho$. Then, there is a non-negative constant β, such that for any $\delta, d > 0$ there exists a pair (G, U) that realizes the $\texttt{HW}_{=1}$-relation of arity $d = |U|$ and cost $\gamma = O(\delta)$, with selection penalty δ and cost tradeoff β. Moreover, such a pair (G, U) together with a path decomposition of G of width $O(d)$ can be computed in time polynomial in $d + \delta$.*

Now, we have all tools at our disposal for our final reduction to replace the $\texttt{HW}_{=1}$-relations.

▶ **Lemma 2.6.** *Let σ be a nonempty finite or simple cofinite set, and ρ be a finite set with $0 \notin \rho$. There is a pathwidth-preserving reduction from (σ, ρ)-$\textsc{ParDomSet}_{\mathcal{R}}$ using only $\texttt{HW}_{=1}$ relations of arity bounded by a constant d to (σ, ρ)-$\textsc{MinParDomSet}$.*

Combining Lemmas 2.2, 2.3, and 2.6 now yields the result of Main Theorem 3 for the case where ρ is finite.

Set ρ is simple cofinite. Unfortunately, we cannot use this approach for the case when ρ is simple cofinite. The problem is that it is *impossible* to build robust realization gadgets for relations of arity 2 or larger when ρ is simple cofinite. Indeed, by the first property of these gadgets (G, U), there is a set S selecting exactly one vertex of U, no vertex of $N(U)$, and exactly γ vertices of $V(G) \setminus U$ that violates no vertex of $V(G) \setminus U$. Because ρ is simple cofinite, we could simply take S, and add all vertices of U to it. This set S' would then still satisfy all vertices of $V(G) \setminus U$ because the unselected vertices of $N(U)$ cannot become violated due to having even more selected neighbors, and clearly S' still selects exactly γ vertices of $V(G) \setminus U$. However, this violates the third property of the definition of these gadgets, which states that any set that selects more than one vertex of U must violate a vertex of $V(G) \setminus U$ or select more than γ vertices of $V(G) \setminus U$. Hence, we need a new approach to handle the case when ρ is simple cofinite.

Step 1. Since we are unable to later replace $\texttt{HW}_{=1}$-relations with gadgets, we cannot work with arbitrary relations for the intermediate problem. Hence, we need to restrict the types of relations we use. A relation R is superset-closed if for any $r \in \mathsf{acc}(R)$ also any superset $r' \subseteq \mathsf{scp}(R)$ of r is in $\mathsf{acc}(R)$. We only allow such relations in the intermediate problem.

(σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$

Input:	Graph G, integers k, ℓ, set of superset-closed relations $\mathcal{R}$
Question:	Is there a set $S \subseteq V(G)$ such that $\lvert S\rvert \leq k$ and at most ℓ vertices of $V(G)$ are violated by S relative to (σ, ρ), and for each $R \in \mathcal{R}$ we have $S \cap \mathsf{scp}(R) \in \mathsf{acc}(R)$?

An instance of (σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$ has arity d (or at most d), when $\lvert\mathsf{scp}(R)\rvert \leq d$ for each $R \in \mathcal{R}$. A path decomposition of an instance of (σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$ is a path decomposition of the input graph G such that additionally for each relation R there is a bag that contains $\mathsf{scp}(R)$.

Even when only allowing superset-closed relations, we still cannot easily replace them with gadgets. We define a technical property of problem instances, being "good", that we additionally exploit to replace superset-closed relations.

▶ **Definition 2.7** (Good Instances). *Let $(G, k, \ell, \mathcal{R})$ be an instance of the problem (σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$. The instance is* good *if any set $S \subseteq V(G)$ that fulfills all relations of $\mathcal{R}$ fulfills*

1. *$\lvert S\rvert \geq k$, and*
2. *there is a set $D \subseteq S$ of size at least ℓ, such that any $v \in D$ has more than $\max \sigma$ neighbors that are also selected by S.*

Note that if σ is cofinite, then $\max \sigma = \infty$ and hence the second condition can be satisfied only if $\ell = 0$, in which case the condition is vacuously true.

We then show the following lower bound for the intermediate problem.

▶ **Lemma 2.8.** *Let σ be a nonempty finite or simple cofinite set, and ρ be a simple cofinite set. For any $\varepsilon > 0$, there is a constant d such that if there is an algorithm that can solve the problem (σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$ on good instances of arity at most d, where a path decomposition of width pw of the instance is provided as part of the input I, in time $(\lvert\mathbb{A}_\mathsf{p}\rvert - \varepsilon)^{\mathsf{pw}} \cdot \lvert I\rvert^{O(1)}$, then the PWSETH is false.*

Step 2. Next, we show that we can reduce to problems that use simpler relations. A $\texttt{HW}_{\geq 1}$-relation accepts a selection from its relation scope if and only if it is nonempty.

▶ **Lemma 2.9.** *There exists a pathwidth-preserving reduction from the problem (σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$ on instances with arity bounded by some constant d to the problem (σ, ρ)-MinParDomSet$_{\mathcal{R}_\supseteq}$ in which each used relation is a $\texttt{HW}_{\geq 1}$-relation of arity at most d. Moreover, if the input instance is a good instance, then the output instance is a good instance.*

Step 3. Finally, we replace $\mathtt{HW}_{\geq 1}$-relations with appropriate gadgets. Ideally, we would like to have gadgets which are similar to the robust realization gadgets we used for the case when ρ is finite. In particular, the gadgets should have some vertex set U such that selecting vertices of $N(U)$ can be made arbitrarily bad. Unfortunately, it is not possible to create such gadgets. For example, if σ and ρ are simple cofinite, then selecting a vertex v compared to not selecting it is never bad for the neighbors of v. Hence, we cannot have a large penalty for selecting vertices of $N(U)$. On the flip side, we can at least ensure that selecting a vertex of $N(U)$, or no vertex of U is somewhat bad: either a vertex of the gadget that is not in U is violated, or the solution size is larger than it needs to be. This intuition is captured by the next definition.

▶ **Definition 2.10** (Fragile Realization)**.** *Let σ and ρ be sets of non-negative integers. A pair (G, U), where G is a graph and $U \subseteq V(G)$ fragilely realizes the $\mathtt{HW}_{\geq 1}$-relation of arity $|U|$ with cost γ, if it has all the following properties:*

1. *Any set $S \subseteq V(G)$ that violates no vertex of $V(G) \setminus U$ fulfills $|S \setminus U| \geq \gamma$.*
2. *For any nonempty $U' \subseteq U$, there is a set $S \subseteq V(G)$ such that $S \cap U = U'$, S violates no vertex of $V(G) \setminus U$, $S \cap N_G(U) = \emptyset$, and $|S \setminus U| = \gamma$.*
3. *Any set $S \subseteq V(G)$ which selects no vertex of U, or selects a vertex of $N_G(U)$ violates a vertex of $V(G) \setminus U$, or has the property that $|S \setminus U| > \gamma$.*
4. *The set U is an independent set of G.*

We construct such gadgets, and then use them in the following reduction. Note that the reduction relies on the input instance being good, otherwise the much weaker guarantees provided by a fragile realization gadget would not be sufficient.

▶ **Lemma 2.11.** *Let σ be a nonempty set, and ρ a simple cofinite set with $0 \notin \rho$. There is a pathwidth-preserving reduction from good instances of (σ, ρ)-*MinParDomSet*$_{\mathcal{R}_{\supseteq}}$ with arity bounded by some constant d using only $\mathtt{HW}_{\geq 1}$-relations to (σ, ρ)-*MinParDomSet*.*

By combining Lemmas 2.8, 2.9, and 2.11, we obtain Main Theorem 3 when the set ρ is simple cofinite.

References

1 Jochen Alber, Hans L. Bodlaender, Henning Fernau, Ton Kloks, and Rolf Niedermeier. Fixed parameter algorithms for DOMINATING SET and related problems on planar graphs. *Algorithmica*, 33(4):461–493, 2002. doi:10.1007/s00453-001-0116-5.

2 Jochen Alber and Rolf Niedermeier. Improved tree decomposition based algorithms for domination-like problems. In Sergio Rajsbaum, editor, *LATIN 2002: Theoretical Informatics, 5th Latin American Symposium, Cancun, Mexico, April 3-6, 2002, Proceedings*, volume 2286 of *Lecture Notes in Computer Science*, pages 613–628. Springer, 2002. doi:10.1007/3-540-45995-2_52.

3 Omid Amini, Fedor V. Fomin, and Saket Saurabh. Implicit branching and parameterized partial cover problems. *J. Comput. Syst. Sci.*, 77(6):1159–1171, 2011. doi:10.1016/j.jcss.2010.12.002.

4 Nikita Andreev, Ivan Bliznets, Madhumita Kundu, Saket Saurabh, Vikash Tripathi, and Shaily Verma. Parameterized complexity of paired domination. In Adele Anna Rescigno and Ugo Vaccaro, editors, *Combinatorial Algorithms – 35th International Workshop, IWOCA 2024, Ischia, Italy, July 1-3, 2024, Proceedings*, volume 14764 of *Lecture Notes in Computer Science*, pages 523–536. Springer, 2024. doi:10.1007/978-3-031-63021-7_40.

5 Stefan Arnborg, Jens Lagergren, and Detlef Seese. Easy problems for tree-decomposable graphs. *J. Algorithms*, 12(2):308–340, 1991. doi:10.1016/0196-6774(91)90006-K.

6 Suman Kalyan Bera, Shalmoli Gupta, Amit Kumar, and Sambuddha Roy. Approximation algorithms for the partition vertex cover problem. *Theor. Comput. Sci.*, 555:2–8, 2014. doi:10.1016/j.tcs.2014.04.006.

7 Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Fourier meets möbius: fast subset convolution. In David S. Johnson and Uriel Feige, editors, *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, pages 67–74. ACM, 2007. doi:10.1145/1250790.1250801.

8 Markus Bläser. Computing small partial coverings. *Inf. Process. Lett.*, 85(6):327–331, 2003. doi:10.1016/S0020-0190(02)00434-9.

9 Hans L. Bodlaender. Treewidth: Algorithmic techniques and results. In Igor Privara and Peter Ruzicka, editors, *MFCS*, volume 1295 of *Lecture Notes in Computer Science*, pages 19–36. Springer, 1997. doi:10.1007/BFB0029946.

10 Hans L. Bodlaender, Sudeshna Kolay, and Astrid Pieterse. Parameterized complexity of conflict-free graph coloring. *SIAM J. Discret. Math.*, 35(3):2003–2038, 2021. doi:10.1137/19M1307160.

11 Hans L. Bodlaender and Arie M. C. A. Koster. Combinatorial optimization on graphs of bounded treewidth. *Comput. J.*, 51(3):255–269, 2008. doi:10.1093/comjnl/bxm037.

12 Hans L. Bodlaender and Arie M. C. A. Koster. Treewidth computations i. upper bounds. *Inf. Comput.*, 208(3):259–275, 2010. doi:10.1016/J.IC.2009.03.008.

13 Hans L. Bodlaender, Erik Jan van Leeuwen, Johan M. M. van Rooij, and Martin Vatshelle. Faster algorithms on branch and clique decompositions. In Petr Hlinený and Antonín Kucera, editors, *Mathematical Foundations of Computer Science 2010, 35th International Symposium, MFCS 2010, Brno, Czech Republic, August 23-27, 2010. Proceedings*, volume 6281 of *Lecture Notes in Computer Science*, pages 174–185. Springer, 2010. doi:10.1007/978-3-642-15155-2_17.

14 Narek Bojikian, Alexander Firbas, Robert Ganian, Hung P. Hoang, and Krisztina Szilágyi. Fine-Grained Complexity of Computing Degree-Constrained Spanning Trees. In Sayan Bhattacharya, Danupon Nanongkai, Michael Benedikt, and Gabriele Puppis, editors, *53rd International Colloquium on Automata, Languages, and Programming (ICALP 2026)*, volume 374 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 38:1–38:14, Dagstuhl, Germany, 2026. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ICALP.2026.38.

15 Glencora Borradaile and Hung Le. Optimal dynamic program for r-domination problems over tree decompositions. In Jiong Guo and Danny Hermelin, editors, *11th International Symposium on Parameterized and Exact Computation, IPEC 2016, August 24-26, 2016, Aarhus, Denmark*, volume 63 of *LIPIcs*, pages 8:1–8:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.IPEC.2016.8.

16 Nader H. Bshouty and Lynn Burroughs. Massaging a linear programming solution to give a 2-approximation for a generalization of the vertex cover problem. In Michel Morvan, Christoph Meinel, and Daniel Krob, editors, *STACS 98, 15th Annual Symposium on Theoretical Aspects of Computer Science, Paris, France, February 25-27, 1998, Proceedings*, volume 1373 of *Lecture Notes in Computer Science*, pages 298–308. Springer, 1998. doi:10.1007/BFb0028569.

17 Binh-Minh Bui-Xuan, Jan Arne Telle, and Martin Vatshelle. Fast dynamic programming for locally checkable vertex subset and vertex partitioning problems. *Theor. Comput. Sci.*, 511:66–76, 2013. doi:10.1016/j.tcs.2013.01.009.

18 Leizhen Cai. Parameterized complexity of cardinality constrained optimization problems. *Comput. J.*, 51(1):102–121, 2008. doi:10.1093/comjnl/bxm086.

19 Chris Calabro, Russell Impagliazzo, and Ramamohan Paturi. The complexity of satisfiability of small depth circuits. In Jianer Chen and Fedor V. Fomin, editors, *Parameterized and Exact Computation, 4th International Workshop, IWPEC 2009, Copenhagen, Denmark, September 10-11, 2009, Revised Selected Papers*, volume 5917 of *Lecture Notes in Computer Science*, pages 75–85. Springer, 2009. doi:10.1007/978-3-642-11269-0_6.

20 Bugra Çaskurlu, Vahan Mkrtchyan, Ojas Parekh, and K. Subramani. On partial vertex cover and budgeted maximum coverage problems in bipartite graphs. In Josep Díaz, Ivan Lanese, and Davide Sangiorgi, editors, *Theoretical Computer Science – 8th IFIP TC 1/WG 2.2 International Conference, TCS 2014, Rome, Italy, September 1-3, 2014. Proceedings*, volume 8705 of *Lecture Notes in Computer Science*, pages 13–26. Springer, 2014. doi:10.1007/978-3-662-44602-7_2.

21 David Cattanéo and Simon Perdrix. The parameterized complexity of domination-type problems and application to linear codes. In T. V. Gopal, Manindra Agrawal, Angsheng Li, and S. Barry Cooper, editors, *Theory and Applications of Models of Computation – 11th Annual Conference, TAMC 2014, Chennai, India, April 11-13, 2014. Proceedings*, volume 8402 of *Lecture Notes in Computer Science*, pages 86–103, Cham, 2014. Springer. doi:10.1007/978-3-319-06089-7_7.

22 Mathieu Chapelle. Parameterized Complexity of Generalized Domination Problems on Bounded Tree-Width Graphs. *CoRR*, abs/1004.2642v5, 2010. doi:10.48550/arXiv.1004.2642.

23 Moses Charikar, Samir Khuller, David M. Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In S. Rao Kosaraju, editor, *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms, January 7-9, 2001, Washington, DC, USA*, pages 642–651. ACM/SIAM, 2001. URL: http://dl.acm.org/citation.cfm?id=365411.365555.

24 Chandra Chekuri, Tanmay Inamdar, Kent Quanrud, Kasturi R. Varadarajan, and Zhao Zhang. Algorithms for covering multiple submodular constraints and applications. *J. Comb. Optim.*, 44(2):979–1010, 2022. doi:10.1007/s10878-022-00874-x.

25 Bruno Courcelle. The monadic second-order logic of graphs. i. recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

26 Radu Curticapean and Dániel Marx. Tight conditional lower bounds for counting perfect matchings on graphs of bounded treewidth, cliquewidth, and genus. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1650–1669. SIAM, 2016. doi:10.1137/1.9781611974331.ch113.

27 Marek Cygan, Holger Dell, Daniel Lokshtanov, Dániel Marx, Jesper Nederlof, Yoshio Okamoto, Ramamohan Paturi, Saket Saurabh, and Magnus Wahlström. On problems as hard as CNF-SAT. *ACM Trans. Algorithms*, 12(3):41:1–41:24, 2016. doi:10.1145/2925416.

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

29 Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Fast hamiltonicity checking via bases of perfect matchings. *J. ACM*, 65(3):12:1–12:46, 2018. doi:10.1145/3148227.

30 Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michal Pilipczuk, Johan M. M. van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. *ACM Trans. Algorithms*, 18(2):17:1–17:31, 2022. doi:10.1145/3506707.

31 Louis Dublois, Michael Lampis, and Vangelis Th. Paschos. New algorithms for mixed dominating set. *Discret. Math. Theor. Comput. Sci.*, 23(1), 2021. doi:10.46298/dmtcs.6824.

32 Louis Dublois, Michael Lampis, and Vangelis Th. Paschos. Upper dominating set: Tight algorithms for pathwidth and sub-exponential approximation. *Theor. Comput. Sci.*, 923:271–291, 2022. doi:10.1016/j.tcs.2022.05.013.

33 László Egri, Dániel Marx, and Pawel Rzazewski. Finding list homomorphisms from bounded-treewidth graphs to reflexive graphs: a complete complexity characterization. In Rolf Niedermeier and Brigitte Vallée, editors, *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, volume 96 of *LIPIcs*, pages 27:1–27:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.STACS.2018.27.

34 Baris Can Esmer, Jacob Focke, Dániel Marx, and Pawel Rzazewski. Fundamental problems on bounded-treewidth graphs: The real source of hardness. In Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, editors, *51st International Colloquium on Automata, Languages, and Programming, ICALP 2024, July 8-12, 2024, Tallinn, Estonia*, volume 297 of *LIPIcs*, pages 34:1–34:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.ICALP.2024.34.

35 Baris Can Esmer, Jacob Focke, Dániel Marx, and Pawel Rzazewski. List homomorphisms by deleting edges and vertices: Tight complexity bounds for bounded-treewidth graphs. In Timothy M. Chan, Johannes Fischer, John Iacono, and Grzegorz Herman, editors, *32nd Annual European Symposium on Algorithms, ESA 2024, September 2-4, 2024, Royal Holloway, London, United Kingdom*, volume 308 of *LIPIcs*, pages 39:1–39:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.ESA.2024.39.

36 Baris Can Esmer and Dániel Marx. Generalized graph packing problems parameterized by treewidth. In Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman, editors, *33rd Annual European Symposium on Algorithms, ESA 2025, Warsaw, Poland, September 15-17, 2025*, volume 351 of *LIPIcs*, pages 3:1–3:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.ESA.2025.3.

37 Omid Etesami, Narges Ghareghani, Michel Habib, Mohammad Reza Hooshmandasl, Reza Naserasr, and Pouyeh Sharifani. When an optimal dominating set with given constraints exists. *Theor. Comput. Sci.*, 780:54–65, 2019. doi:10.1016/j.tcs.2019.02.012.

38 Jacob Focke, Dániel Marx, Fionn Mc Inerney, Daniel Neuen, Govind S. Sankar, Philipp Schepper, and Philip Wellnitz. Tight complexity bounds for counting generalized dominating sets in bounded-treewidth graphs – part I: algorithmic results. *ACM Trans. Algorithms*, 21(3):27:1–27:45, 2025. doi:10.1145/3731452.

39 Jacob Focke, Dániel Marx, Fionn Mc Inerney, Daniel Neuen, Govind S. Sankar, Philipp Schepper, and Philip Wellnitz. Tight complexity bounds for counting generalized dominating sets in bounded-treewidth graphs part II: Hardness results. *ACM Trans. Comput. Theory*, 17(2), April 2025. doi:10.1145/3708509.

40 Jacob Focke, Dániel Marx, and Pawel Rzazewski. Counting list homomorphisms from graphs of bounded treewidth: Tight complexity bounds. *ACM Trans. Algorithms*, 20(2):11, 2024. doi:10.1145/3640814.

41 Fedor V. Fomin, Petr A. Golovach, Jan Kratochvíl, Dieter Kratsch, and Mathieu Liedloff. Sort and search: Exact algorithms for generalized domination. *Inf. Process. Lett.*, 109(14):795–798, June 2009. doi:10.1016/j.ipl.2009.03.023.

42 Fedor V. Fomin, Petr A. Golovach, Jan Kratochvíl, Dieter Kratsch, and Mathieu Liedloff. Branch and recharge: Exact algorithms for generalized domination. *Algorithmica*, 61(2):252–273, 2011. doi:10.1007/s00453-010-9418-9.

43 Fedor V. Fomin and Dieter Kratsch. *Exact Exponential Algorithms*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2010. doi:10.1007/978-3-642-16533-7.

44 Fedor V. Fomin, Daniel Lokshtanov, Venkatesh Raman, and Saket Saurabh. Subexponential algorithms for partial cover problems. *Inf. Process. Lett.*, 111(16):814–818, 2011. doi:10.1016/j.ipl.2011.05.016.

45 Rajiv Gandhi, Samir Khuller, and Aravind Srinivasan. Approximation algorithms for partial covering problems. *J. Algorithms*, 53(1):55–84, 2004. doi:10.1016/j.jalgor.2004.04.002.

46 Petr A. Golovach and Jan Kratochvíl. Computational complexity of generalized domination: A complete dichotomy for chordal graphs. In Andreas Brandstädt, Dieter Kratsch, and Haiko Müller, editors, *Graph-Theoretic Concepts in Computer Science, 33rd International Workshop, WG 2007, Dornburg, Germany, June 21-23, 2007. Revised Papers*, volume 4769 of *Lecture Notes in Computer Science*, pages 1–11. Springer, 2007. doi:10.1007/978-3-540-74839-7_1.

47 Petr A. Golovach, Jan Kratochvíl, and Ondrej Suchý. Parameterized complexity of generalized domination problems. *Discret. Appl. Math.*, 160(6):780–792, 2012. doi:10.1016/j.dam.2010.11.012.

48 Petr A. Golovach and Yngve Villanger. Parameterized complexity for domination problems on degenerate graphs. In Hajo Broersma, Thomas Erlebach, Tom Friedetzky, and Daniël Paulusma, editors, *Graph-Theoretic Concepts in Computer Science, 34th International Workshop, WG 2008, Durham, UK, June 30 – July 2, 2008. Revised Papers*, volume 5344 of *Lecture Notes in Computer Science*, pages 195–205, 2008. doi:10.1007/978-3-540-92248-3_18.

49 Jakob Greilhuber, Philipp Schepper, and Philip Wellnitz. Residue domination in bounded-treewidth graphs. In Olaf Beyersdorff, Michal Pilipczuk, Elaine Pimentel, and Kim Thang Nguyen, editors, *42nd International Symposium on Theoretical Aspects of Computer Science, STACS 2025, March 4-7, 2025, Jena, Germany*, volume 327 of *LIPIcs*, pages 41:1–41:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.STACS.2025.41.

50 Carla Groenland, Isja Mannens, Jesper Nederlof, Marta Piecyk, and Pawel Rzazewski. Towards tight bounds for the graph homomorphism problem parameterized by cutwidth via asymptotic matrix parameters. In Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, editors, *51st International Colloquium on Automata, Languages, and Programming, ICALP 2024, July 8-12, 2024, Tallinn, Estonia*, volume 297 of *LIPIcs*, pages 77:1–77:21. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.ICALP.2024.77.

51 Jiong Guo, Rolf Niedermeier, and Sebastian Wernicke. Parameterized complexity of generalized vertex cover problems. In Frank K. H. A. Dehne, Alejandro López-Ortiz, and Jörg-Rüdiger Sack, editors, *Algorithms and Data Structures, 9th International Workshop, WADS 2005, Waterloo, Canada, August 15-17, 2005, Proceedings*, volume 3608 of *Lecture Notes in Computer Science*, pages 36–48. Springer, 2005. doi:10.1007/11534273_5.

52 Magnús M. Halldórsson, Jan Kratochvíl, and Jan Arne Telle. Independent sets with domination constraints. *Discret. Appl. Math.*, 99(1-3):39–54, 2000. doi:10.1016/S0166-218X(99)00124-9.

53 Eran Halperin and Aravind Srinivasan. Improved approximation algorithms for the partial vertex cover problem. In Klaus Jansen, Stefano Leonardi, and Vijay V. Vazirani, editors, *Approximation Algorithms for Combinatorial Optimization, 5th International Workshop, APPROX 2002, Rome, Italy, September 17-21, 2002, Proceedings*, volume 2462 of *Lecture Notes in Computer Science*, pages 161–174. Springer, 2002. doi:10.1007/3-540-45753-4_15.

54 Tesshu Hanaka, Ioannis Katsikarelis, Michael Lampis, Yota Otachi, and Florian Sikora. Parameterized orientable deletion. *Algorithmica*, 82(7):1909–1938, 2020. doi:10.1007/s00453-020-00679-6.

55 Tim A. Hartmann and Dániel Marx. Independence and domination on bounded-treewidth graphs: Integer, rational, and irrational distances. In Olaf Beyersdorff, Michal Pilipczuk, Elaine Pimentel, and Kim Thang Nguyen, editors, *42nd International Symposium on Theoretical Aspects of Computer Science, STACS 2025, March 4-7, 2025, Jena, Germany*, volume 327 of *LIPIcs*, pages 44:1–44:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.STACS.2025.44.

56 Falko Hegerfeld and Stefan Kratsch. Towards exact structural thresholds for parameterized complexity. In Holger Dell and Jesper Nederlof, editors, *17th International Symposium on Parameterized and Exact Computation, IPEC 2022, September 7-9, 2022, Potsdam, Germany*, volume 249 of *LIPIcs*, pages 17:1–17:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.IPEC.2022.17.

57 Pinar Heggernes and Jan Arne Telle. Partitioning graphs into generalized dominating sets. *Nord. J. Comput.*, 5(2):128–142, 1998.

58 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

59 Tanmay Inamdar and Kasturi R. Varadarajan. On partial covering for geometric set systems. In Bettina Speckmann and Csaba D. Tóth, editors, *34th International Symposium on Computational Geometry, SoCG 2018, June 11-14, 2018, Budapest, Hungary*, volume 99 of *LIPIcs*, pages 47:1–47:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.SOCG.2018.47.

60 Toshimasa Ishii, Hirotaka Ono, and Yushi Uno. Subexponential fixed-parameter algorithms for partial vector domination. *Discret. Optim.*, 22:111–121, 2016. doi:10.1016/j.disopt.2016.01.003.

61 Lars Jaffke, O-joung Kwon, Torstein J. F. Strømme, and Jan Arne Telle. Mim-width III. graph powers and generalized distance domination problems. *Theor. Comput. Sci.*, 796:216–236, 2019. doi:10.1016/j.tcs.2019.09.012.

62 Bart M. P. Jansen and Jesper Nederlof. Computing the chromatic number using graph decompositions via matrix rank. *Theor. Comput. Sci.*, 795:520–539, 2019. doi:10.1016/J.TCS.2019.08.006.

63 Ioannis Katsikarelis, Michael Lampis, and Vangelis Th. Paschos. Structurally parameterized d-scattered set. *Discret. Appl. Math.*, 308:168–186, 2022. doi:10.1016/j.dam.2020.03.052.

64 Joachim Kneis, Daniel Mölle, and Peter Rossmanith. Partial vs. complete domination: t-dominating set. In Jan van Leeuwen, Giuseppe F. Italiano, Wiebe van der Hoek, Christoph Meinel, Harald Sack, and Frantisek Plásil, editors, *SOFSEM 2007: Theory and Practice of Computer Science, 33rd Conference on Current Trends in Theory and Practice of Computer Science, Harrachov, Czech Republic, January 20-26, 2007, Proceedings*, volume 4362 of *Lecture Notes in Computer Science*, pages 367–376. Springer, 2007. doi:10.1007/978-3-540-69507-3_31.

65 Ioannis Koutis and Ryan Williams. LIMITS and applications of group algebras for parameterized problems. *ACM Trans. Algorithms*, 12(3):31:1–31:18, 2016. doi:10.1145/2885499.

66 Michael Lampis. Finer tight bounds for coloring on clique-width. *SIAM J. Discret. Math.*, 34(3):1538–1558, 2020. doi:10.1137/19M1280326.

67 Michael Lampis. The primal pathwidth SETH. In Yossi Azar and Debmalya Panigrahi, editors, *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2025, New Orleans, LA, USA, January 12-15, 2025*, pages 1494–1564. SIAM, 2025. doi:10.1137/1.9781611978322.47.

68 Michael Lampis and Manolis Vasilakis. Structural parameterizations for two bounded degree problems revisited. *ACM Trans. Comput. Theory*, 16(3):17:1–17:51, 2024. doi:10.1145/3665156.

69 Michael Lampis and Manolis Vasilakis. Structural parameterizations for induced and acyclic matching. In Henning Fernau and Philipp Kindermann, editors, *Graph-Theoretic Concepts in Computer Science – 51st International Workshop, WG 2025, Otzenhausen, Germany, June 11-13, 2025, Revised Selected Papers*, volume 16124 of *Lecture Notes in Computer Science*, pages 360–376. Springer, 2025. doi:10.1007/978-3-032-11835-6_26.

70 Ke Liu and Mei Lu. w-dominating set problem on graphs of bounded treewidth. *arXiv*, 2021. doi:10.48550/arXiv.2101.02867.

71 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Known algorithms on graphs of bounded treewidth are probably optimal. *ACM Trans. Algorithms*, 14(2):13:1–13:30, 2018. doi:10.1145/3170442.

72 Pasin Manurangsi. A note on max k-vertex cover: Faster fpt-as, smaller approximate kernel and improved approximation. In Jeremy T. Fineman and Michael Mitzenmacher, editors, *2nd Symposium on Simplicity in Algorithms, SOSA 2019, January 8-9, 2019, San Diego, CA, USA*, volume 69 of *OASIcs*, pages 15:1–15:21. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/OASIcs.SOSA.2019.15.

73 Dániel Marx, Govind S. Sankar, and Philipp Schepper. Degrees and gaps: Tight complexity results of general factor problems parameterized by treewidth and cutwidth. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference)*, volume 198 of *LIPIcs*, pages 95:1–95:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ICALP.2021.95.

74 Dániel Marx, Govind S. Sankar, and Philipp Schepper. Anti-factor is FPT parameterized by treewidth and list size (but counting is hard). *Algorithmica*, 87(1):22–88, 2025. doi:10.1007/s00453-024-01265-w.

75 Tomás Masarík, Jedrzej Olkowski, and Anna Zych-Pawlewicz. Fair vertex problems parameterized by cluster vertex deletion. *CoRR*, abs/2502.01400, 2025. doi:10.48550/arXiv.2502.01400.

76 Richard Matthew McCutchen and Samir Khuller. Streaming algorithms for k-center clustering with outliers and with anonymity. In Ashish Goel, Klaus Jansen, José D. P. Rolim, and Ronitt Rubinfeld, editors, *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques, 11th International Workshop, APPROX 2008, and 12th International Workshop, RANDOM 2008, Boston, MA, USA, August 25-27, 2008. Proceedings*, volume 5171 of *Lecture Notes in Computer Science*, pages 165–178. Springer, 2008. doi:10.1007/978-3-540-85363-3_14.

77 Mohsen Alambardar Meybodi, Fedor V. Fomin, Amer E. Mouawad, and Fahad Panolan. On the parameterized complexity of $[1, j]$-domination problems. *Theor. Comput. Sci.*, 804:207–218, 2020. doi:10.1016/j.tcs.2019.11.032.

78 Karolina Okrasa, Marta Piecyk, and Pawel Rzazewski. Full complexity classification of the list homomorphism problem for bounded-treewidth graphs. In Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders, editors, *28th Annual European Symposium on Algorithms, ESA 2020, September 7-9, 2020, Pisa, Italy (Virtual Conference)*, volume 173 of *LIPIcs*, pages 74:1–74:24. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ESA.2020.74.

79 Karolina Okrasa and Pawel Rzazewski. Fine-grained complexity of the graph homomorphism problem for bounded-treewidth graphs. *SIAM J. Comput.*, 50(2):487–508, 2021. doi:10.1137/20M1320146.

80 Mehdy Roayaei and Mohammadreza Razzazi. An fpt-algorithm for modifying a graph of bounded treewidth to decrease the size of its dominating set using minimum modification. *Inf. Process. Lett.*, 116(9):590–594, 2016. doi:10.1016/j.ipl.2016.04.002.

81 Neil Robertson and P. D. Seymour. Graph minors. II. algorithmic aspects of tree-width. *Journal of Algorithms*, 7(3):309–322, 1986. doi:10.1016/0196-6774(86)90023-4.

82 Philipp Schepper. *Faster, Higher, Easier: Toward a Systematic Study of Parameterized Vertex and Edge Selection Problems*. PhD thesis, Saarland University, Germany, 2025. doi:10.22028/D291-46499.

83 Petr Slavík. Improved performance of the greedy algorithm for partial cover. *Inf. Process. Lett.*, 64(5):251–254, 1997. doi:10.1016/S0020-0190(97)00182-8.

84 Jan Arne Telle. Complexity of domination-type problems in graphs. *Nord. J. Comput.*, 1(1):157–171, 1994.

85 Jan Arne Telle and Andrzej Proskurowski. Practical algorithms on partial k-trees with an application to domination-like problems. In Frank K. H. A. Dehne, Jörg-Rüdiger Sack, Nicola Santoro, and Sue Whitesides, editors, *Algorithms and Data Structures, Third Workshop, WADS '93, Montréal, Canada, August 11-13, 1993, Proceedings*, volume 709 of *Lecture Notes in Computer Science*, pages 610–621. Springer, 1993. doi:10.1007/3-540-57155-8_284.

86 Jan Arne Telle and Andrzej Proskurowski. Algorithms for vertex partitioning problems on partial k-trees. *SIAM J. Discret. Math.*, 10(4):529–550, November 1997. doi:10.1137/S0895480194275825.

87 Johan M. M. van Rooij. Fast algorithms for join operations on tree decompositions. In Fedor V. Fomin, Stefan Kratsch, and Erik Jan van Leeuwen, editors, *Treewidth, Kernels, and Algorithms – Essays Dedicated to Hans L. Bodlaender on the Occasion of His 60th Birthday*, volume 12160 of *Lecture Notes in Computer Science*, pages 262–297. Springer, 2020. doi:10.1007/978-3-030-42071-0_18.

88 Johan M. M. van Rooij. A generic convolution algorithm for join operations on tree decompositions. In Rahul Santhanam and Daniil Musatov, editors, *Computer Science – Theory and Applications – 16th International Computer Science Symposium in Russia, CSR 2021, Sochi, Russia, June 28 – July 2, 2021, Proceedings*, volume 12730 of *Lecture Notes in Computer Science*, pages 435–459. Springer, 2021. doi:10.1007/978-3-030-79416-3_27.

89 Johan M. M. van Rooij, Hans L. Bodlaender, and Peter Rossmanith. Dynamic programming on tree decompositions using generalised fast subset convolution. In Amos Fiat and Peter Sanders, editors, *Algorithms – ESA 2009, 17th Annual European Symposium, Copenhagen, Denmark, September 7-9, 2009. Proceedings*, volume 5757 of *Lecture Notes in Computer Science*, pages 566–577. Springer, 2009. doi:10.1007/978-3-642-04128-0_51.

90 C. D. T. Watson-Gandy. Heuristic procedures for the m-partial cover problem on a plane. *European Journal of Operational Research*, 11(2):149–157, 1982. doi:10.1016/0377-2217(82)90109-6.

91 Virginia Vassilevska Williams. On some fine-grained questions in algorithms and complexity. In *Proceedings of the International Congress of Mathematicians (ICM 2018)*, volume 4, pages 3447–3487, 2019. doi:10.1142/9789813272880_0188.