Mim-Width III. Graph Powers and Generalized Distance Domination Problems*

Lars Jaffke[†1], O-joung Kwon[‡2,3], Torstein J. F. Strømme[1], and Jan Arne Telle[1]

[1]Department of Informatics, University of Bergen, Norway.
{lars.jaffke, torstein.stromme, jan.arne.telle}@uib.no
[2]Department of Mathematics, Incheon National University, South Korea.
[3]Discrete Mathematics Group, Institute for Basic Science (IBS), Daejeon, South Korea.
ojoungkwon@gmail.com

September 4, 2019

Abstract

We generalize the family of (σ, ρ) problems and locally checkable vertex partition problems to their distance versions, which naturally captures well-known problems such as Distance-r Dominating Set and Distance-r Independent Set. We show that these distance problems are in XP parameterized by the structural parameter *mim-width*, and hence polynomial-time solvable on graph classes where mim-width is bounded and quickly computable, such as k-trapezoid graphs, Dilworth k-graphs, (circular) permutation graphs, interval graphs and their complements, convex graphs and their complements, k-polygon graphs, circular arc graphs, complements of d-degenerate graphs, and H-graphs if given an H-representation. We obtain these results by showing that taking any power of a graph never increases its mim-width by more than a factor of two. To supplement these findings, we show that many classes of (σ, ρ) problems are W[1]-hard parameterized by mim-width + solution size.

We show that powers of graphs of tree-width $w - 1$ or path-width w and powers of graphs of clique-width w have mim-width at most w. These results provide new classes of bounded mim-width. We prove a slight strengthening of the first statement which implies that, surprisingly, Leaf Power graphs which are of importance in the field of phylogenetic studies have mim-width at most 1.

*The paper is based on extended abstracts that appeared in STACS 2018 [20] and IPEC 2018 [16]. The first part of the series, 'Mim-Width I. Induced Path Problems' [18], and the second part of the series, 'Mim-Width II. The Feedback Vertex Set Problem' [19], are based on extended abstracts that appeared in IPEC 2017 [17] and STACS 2018 [20].

†Supported by the Bergen Research Foundation (BFS).

‡Supported by IBS-R029-C1, and the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education (No. NRF-2018R1D1A1B07050294), and the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (ERC consolidator grant DISTRUCT, agreement No. 648527). Part of the research took place while Kwon was at Logic and Semantics, Technische Universität Berlin, Berlin, Germany.

1 Introduction

Telle and Proskurowski [30] defined the (σ, ρ)-domination problems, or (σ, ρ) problems for short, and the more general *locally checkable vertex partitioning* problems (LCVP). In (σ, ρ)-domination problems, feasible solutions are vertex sets with constraints on how many neighbors each vertex of the graph has in the set. The framework generalizes important and well-studied problems such as Maximum Independent Set and Minimum Dominating Set, as well as Perfect Code, Minimum subgraph with minimum degree d and a multitude of other problems. See Table 1. Bui-Xuan, Telle and Vatshelle [7] showed that (σ, ρ)-domination and locally checkable vertex partitioning problems can be solved in time XP parameterized by *mim-width*, if we are given a corresponding decomposition tree. Roughly speaking, the structural parameter mim-width measures how easy it is to decompose a graph along vertex cuts inducing a bipartite graph with small maximum induced matching size [31].

In this paper, we consider distance versions of problems related to independence and domination, such as Distance-r Independent Set and Distance-r Dominating Set. The Distance-r Independent Set problem, also studied under the names r-Scattered Set and r-Dispersion (see e.g. [2] and the references therein), asks to find a set of at least k vertices whose vertices have pairwise distance strictly longer than r. Agnarsson et al. [1] pointed out that it is identical to the original Independent Set problem on the r-th power[1] graph G^r of the input graph G, and also showed that for fixed r, it can be solved in linear time for interval graphs, and circular arc graphs. The Distance-r Dominating Set problem was introduced by Slater [29] and Henning et al. [15]. They as well observed that it is identical to solve the original Dominating Set problem on the r-th power graph. Slater presented a linear-time algorithm to solve Distance-r Dominating Set problem on forests.

We generalize all of the (σ, ρ)-domination and LCVP problems to their distance versions, which naturally captures Distance-r Independent Set and Distance-r Dominating Set. While the original problems put constraints on the size of the immediate neighborhood of a vertex, we consider the constraints to be applied to the ball of radius r around it. Consider for instance the Minimum Subgraph with Minimum Degree d problem; where the original problem is asking for the smallest (in terms of number of vertices) subgraph of minimum degree d, we are instead looking for the smallest subgraph such that for each vertex there are at least d vertices at distance at least 1 and at most r. In the Perfect Code problem, the target is to choose a subset of vertices such that each vertex has exactly one chosen vertex in its closed neighborhood. In the distance-r version of the problem, we replace the closed neighborhood by the closed r-neighborhood. This problem is known as Perfect r-Code, and was introduced by Biggs [4] in 1973.

We show that all these distance problems are in XP parameterized by mim-width if a decomposition tree is given. One of the main results of the paper is of structural nature, namely that for any positive integer r the mim-width of a graph power G^r is at most twice the mim-width of G. It follows that we can reduce the distance-r version of a (σ, ρ)-domination problem to its non-distance variant by taking the graph power G^r, while preserving small mim-width.

One negative aspect of the mim-width parameter is that we do not know an XP algorithm computing mim-width. The problem of deciding whether a given graph has a decomposition tree of mim-width w is NP-complete in general and W[1]-hard parameterized by w. Determining the

[1] For a graph G and an integer r, the *r-th power of G*, denoted by G^r, is the graph that has the same vertex set as G, with each pair of distinct vertices being adjacent if their distance in G is at most r.

optimal mim-width is not in APX unless NP = ZPP, making it unlikely to have a polynomial-time constant-factor approximation algorithm [28].

However, for several graph classes we are able to find a decomposition tree of constant mim-width in polynomial time, using the results of Belmonte and Vatshelle [3]. These include; permutation graphs, convex graphs and their complements, interval graphs and their complements (all of which have *linear* mim-width 1); (circular k-) trapezoid graphs, circular permutation graphs, Dilworth-k graphs, k-polygon graphs, circular arc graphs and complements of d-degenerate graphs. Fomin, Golovach and Raymond [13] showed that we can find linear decomposition trees of constant mim-width for the very general class of H-graphs in polynomial time, *given* an H-representation of the input graph.[2] For all of the above graph classes, our results imply that the distance-r (σ, ρ)-domination and LCVP problems are polynomial time solvable.

Graphs represented by intersections of objects in some model are often closed under taking powers. For instance, interval graphs, and generally d-trapezoid graphs [1, 12], circular arc graphs [1, 27], and leaf power graphs (by definition) are such graphs. We refer to [5, Chapter 10.6] for a survey of such results. For these classes, we already know that the distance-r version of a (σ, ρ)-domination problem can be solved in polynomial time. However, this closure property does not always hold; for instance, permutation graphs are not closed under taking powers. Our result implies that to obtain such algorithmic results, we do not need to know that these classes are closed under taking powers; it is sufficient to know that classes have bounded mim-width. To the best of our knowledge, for the most well-studied distance-r (σ, ρ)-domination problem, Distance-r Dominating Set, we obtain the first polynomial time algorithms on Dilworth k-graphs, convex graphs and their complements, complements of interval graphs, k-polygon graphs, H-graphs (given an H-representation of the input graph), and complements of d-degenerate graphs.

We give results that expand our knowledge of the expressive power of mim-width. We show that powers of graphs of tree-width $w - 1$ or path-width w and powers of graphs of clique-width w have mim-width at most w. In fact, the statement we prove is a slight strengthening, namely: Given a nice tree decomposition of width w, all of whose join bags have size at most w, or a clique-width w-expression of a graph, one can output a decomposition tree of mim-width w of its k-th power in polynomial time. The former implies that leaf power graphs, of importance in the field of phylogenetic studies [8], have mim-width 1. These graphs are known to be strongly chordal and there has recently been interest in delineating the difference between strongly chordal graphs and leaf power graphs, on the assumption that this difference was not very big [22, 24]. Our result actually implies a large difference, as it was recently shown by Mengel that there are strongly chordal split graphs of mim-width linear in the number of vertices [23].

The natural question to ask after obtaining an XP algorithm is whether we can do better, e. g. can we show that for all fixed r, the distance-r (σ, ρ)-domination problems are in FPT? Fomin et al. [13] answered this in the negative by showing that (the standard, i.e. distance-1 variants of) Maximum Independent Set, Minimum Dominating Set and Minimum Independent Dominating Set problems are W[1]-hard parameterized by (linear) mim-width + solution size. We modify their reductions to extend these results to several families of (σ, ρ)-domination problems, including the maximization variants of Induced Matching, Induced d-Regular Subgraph and Induced Subgraph of Max Degree $\leq d$, the minimization variants of Total Dominating Set and d-Dominating Set and both the maximization and the minimization variant of Dominating

[2] For a formal definition of H-graphs, see Definition 18. We would like to remark that it is NP-complete to decide whether a graph is an H-graph whenever H is not a cactus [9].

Induced Matching.

The remainder of the paper is organized as follows. In Section 2 we introduce the (σ, ρ)-domination problems and define their distance-r generalization, and we give a short introduction to mim-width. In Section 3 we show that the mim-width of a graph grows by at most a factor 2 when taking (arbitrary large) powers and prove bounds on the mim-width of powers of graphs of bounded tree-width and clique-width. We discuss algorithmic consequences for (σ, ρ) problems and more generally, LCVP problems, in Section 4, and in Section 5 we present the above mentioned lower bounds. Finally, we give some concluding remarks in Section 6.

2 The Main Concepts

In this section, we introduce the main concepts of the paper. That is, in Section 2.2 we introduce the family of distance-r (σ, ρ)-domination problems, and in Section 2.3 we give a short introduction to the graph parameter mim-width and several of its aspects that are of importance to this work. Before we proceed, we introduce the basic terminology and notation used throughout the paper.

2.1 Preliminaries

We let the set of natural numbers be $\mathbb{N} = \{0, 1, 2, \ldots\}$, and the positive natural numbers be $\mathbb{N}^+ = \mathbb{N} \setminus \{0\}$. For a set S and a given property ψ, we denote by S_ψ the biggest subset of S where ψ is satisfied for all elements. For instance, $\mathbb{N}^+_{\leq k}$ denotes the set $\{1, 2, \ldots k\}$. For this particular property, we also use the shorthand $[k] \coloneqq \mathbb{N}^+_{\leq k}$. For a set X, we denote by $\binom{X}{k}$ the set of all size-k subsets of X.

Basic Notation for Graphs. A graph G is a pair of a vertex set $V(G)$ and an edge set $E(G) \subseteq \binom{V(G)}{2}$. For an edge $\{u, v\} \in E(G)$, we use the shorthand notation 'uv'. For a set of edges $F \subseteq E(G)$, we denote by $V(F)$ the set of vertices that are contained in the edges of F, i.e. $V(F) \coloneqq \bigcup_{uv \in F} \{u, v\}$. A *cut* of a graph is a 2-partition of its vertex set.

(Induced) Subgraphs. For graphs G and H we say that H is a *subgraph* of G, denoted by $H \subseteq G$, if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For a vertex set X, we denote by $G[X]$ the subgraph of G *induced* by X, i.e. $G[X] \coloneqq (X, E(G) \cap \binom{X}{2})$. We use the notation $G - X \coloneqq G[V(G) \setminus X]$ and for a set of edges $F \subseteq E(G)$, $G - F \coloneqq (V(G), E(G) \setminus F)$. For a vertex $v \in V(G)$ (an edge $e \in E(G)$), we use the shorthand $G - x \coloneqq G - \{x\}$ $(G - e \coloneqq G - \{e\})$.

Neighborhoods. Let G be a graph. For a vertex $v \in V(G)$, we denote by $N_G(v)$ the *open neighborhood* of v, i.e. $N_G(v) \coloneqq \{w \mid vw \in E(G)\}$, and by $N_G[v]$ the *closed neighborhood* of v, i.e. $N_G[v] \coloneqq \{v\} \cup N_G(v)$. The *degree* of v is the size of its open neighborhood, i.e. $\deg_G(v) \coloneqq |N_G(v)|$. For a set of vertices $X \subseteq V(G)$, we let $N_G(X) \coloneqq \bigcup_{v \in X} N_G(x) \setminus X$ and $N_G[X] \coloneqq N_G(X) \cup X$. We use the shorthand notations 'N' and 'deg' for 'N_G' and '$\deg_G$', respectively, if G is clear from the context.

Connectivity/Distance. A graph G is called *connected* if for each 2-partition (X, Y) of $V(G)$ with $X \neq \emptyset$ and $Y \neq \emptyset$, there is an edge $xy \in E(G)$ such that $x \in X$ and $y \in Y$. A connected graph all of whose vertices have degree two is called a *cycle*. A graph that does not contain a cycle as a subgraph is called a *forest* and a connected forest is a *tree*. The vertices of degree one in a tree are called the *leaves* and the remaining vertices the *internal* vertices. A tree of maximum degree two is called a *path* and the leaves of a path are called *endpoints*. A tree T is called a *caterpillar* if it contains a subgraph $P \subseteq T$ such that P is a path and each vertex in $V(T) \setminus V(P)$ has a neighbor in $V(P)$. The *length* of a path P is $|E(P)|$, i.e. the number of its edges. Given a graph G and two of its vertices $u, v \in V(G)$, the *distance* from u to v in G, denoted by $\textsc{dist}_G(u, v)$, is the smallest length of any path $P \subseteq G$ with endpoints u and v.

Matchings. A set of edges $M \subseteq E(G)$ is called a *matching*, if no pair of edges in M shares an endpoint. A matching is called *induced* if $G[V(M)] = (V(M), M)$, i.e. if there are no other edges than the edges in M in the subgraph of G induced by $V(M)$.

Cliques/Independence/Domination/Bipartite Graphs. A graph G is called *complete*, if $E(G) = \binom{V(G)}{2}$. A set of vertices $C \subseteq V(G)$ is called a *clique* if $G[C]$ is a complete graph. A set of vertices $I \subseteq V(G)$ is called an *independent set* if $E(G[I]) = \emptyset$. A set of vertices $D \subseteq V(G)$ is called a *dominating set* if $N_G[D] = V(G)$. A graph G is called *bipartite* if there is a 2-partition (X, Y) of $V(G)$ such that X and Y are independent sets, and we call a bipartite graph G with partition (X, Y) *complete bipartite* if $E(G) = \{xy \mid x \in X, y \in Y\}$. For two disjoint vertex sets X, $Y \subseteq V(G)$, we denote by $G[X, Y]$ the *bipartite subgraph of* G *induced by* (X, Y), i.e. $G[X, Y] \coloneqq (X \cup Y, E(G) \cap \{xy \mid x \in X, y \in Y\})$.

Multigraphs/Subdivisions. Let G and H be two graphs. A bijection $\varphi\colon V(G) \to V(H)$ is called an *isomorphism from* G *to* H if for all $u, v \in E(G)$, $uv \in E(G)$ if and only if $\varphi(u)\varphi(v) \in E(H)$. If there is an isomorphism from G to H then say that G and H are *isomorphic*.

A *multigraph* is a pair of a set of vertices $V(G)$ and a *multiset* of size-2 subsets of $V(G)$, called the edges of G and denoted by $E(G)$. For a (multi-) graph H, we let $|H| \coloneqq |V(G)|$ and $||H|| \coloneqq |E(G)|$. Let G be a (multi-) graph. An *edge subdivision* of an edge $e \in E(G)$ with endpoints u and v is the operation of adding to G a new vertex x and replacing the edge e by a path with endpoints u and v, consisting of an edge between u and x and an edge between x and v. We call the vertex x a *subdivision vertex*. A graph G' is called a *subdivision of* G if it can be obtained from G by a series of edge subdivisions.

2.2 Distance-r (σ, ρ)-Domination Problems

Let σ and ρ be finite or co-finite subsets of the natural numbers $\sigma, \rho \subseteq \mathbb{N}$. For a graph G, a vertex set $S \subseteq V(G)$ is a (σ, ρ)-*dominating set*, or simply (σ, ρ) *set* if

- for each vertex $v \in S$ it holds that $|N(v) \cap S| \in \sigma$, and
- for each vertex $v \in V(G) \setminus S$ it holds that $|N(v) \cap S| \in \rho$.

For instance, a $(\{0\}, \mathbb{N})$ set is an independent set as there are no edges between the vertices in the set, and we do not care about adjacencies between S and $V(G) \setminus S$. For another example, a $(\mathbb{N}, \mathbb{N}^+)$-set is a dominating set as we require that for each vertex in $V(G) \setminus S$, it has at least one neighbor in S.

There are three types of (σ, ρ)-domination problems: minimization, maximization, and existence. We denote the problem of finding a minimum (maximum) (σ, ρ) set as the Min-(σ, ρ) Domination (Max-(σ, ρ) Domination) problem, or simply Min-(σ, ρ) (Max-(σ, ρ)) problem, and we refer to Table 1 for examples of well-studied problems expressible in this framework.

We naturally generalize (σ, ρ)-domination problems to their distance-r version: For a graph G and a vertex $v \in V(G)$, let $N_G^r(v)$ denote the *ball of radius* r *around* v, i. e.

$$N_G^r(v) \coloneqq \{w \in V(G) \setminus \{v\} \mid \textsc{dist}_G(v, w) \leq r\}.$$

σ	ρ	d	Standard name	W[1]-Hard
$\{0\}$	$\mathbb{N}$	1	Independent set	$\circ_{\max}$
$\mathbb{N}$	$\mathbb{N}^+$	1	Dominating set	$\circ_{\min}$
$\{0\}$	$\mathbb{N}^+$	1	Maximal Independent set	$\circ_{\min}$
$\mathbb{N}^+$	$\mathbb{N}^+$	1	Total Dominating set	$\star_{\min}$
$\{0\}$	$\{0,1\}$	2	Strong Stable set or 2-Packing	
$\{0\}$	$\{1\}$	2	Perfect Code or Efficient Dom. set	
$\{0,1\}$	$\{0,1\}$	2	Total Nearly Perfect set	
$\{0,1\}$	$\{1\}$	2	Weakly Perfect Dominating set	
$\{1\}$	$\{1\}$	2	Total Perfect Dominating set	
$\{1\}$	$\mathbb{N}$	2	Induced Matching	$\star_{\max}$
$\{1\}$	$\mathbb{N}^+$	2	Dominating Induced Matching	$\star_{\max}$, $\star_{\min}$
$\mathbb{N}$	$\{1\}$	2	Perfect Dominating set	
$\mathbb{N}$	$\{d,d+1,...\}$	d	d-Dominating set	$\star_{\min}$
$\{d\}$	$\mathbb{N}$	$d+1$	Induced d-Regular Subgraph	$\star_{\max}$
$\{d,d+1,...\}$	$\mathbb{N}$	d	Subgraph of Min Degree $\geq d$	
$\{0,1,...,d\}$	$\mathbb{N}$	$d+1$	Induced Subg. of Max Degree $\leq d$	$\star_{\max}$

Table 1: Some vertex subset properties expressible as (σ,ρ)-sets, with $\mathbb{N} = \{0,1,...\}$ and $\mathbb{N}^+ = \{1,2,...\}$. Column d shows $d = \max(d(\sigma), d(\rho))$. For each problem, at least one of the minimization, the maximization and the existence problem is NP-complete. For problems marked with $\star_{\max}$ ($\star_{\min}$) in the rightmost column, W[1]-hardness of the maximization (minimization) problem parameterized by mim-width + solution size is shown in the present paper. For problems marked with $\circ_{\max}$ ($\circ_{\min}$) the W[1]-hardness of maximization (minimization) in the same parameterization was shown by Fomin et al. [13].

Let σ and ρ be finite or co-finite subsets of $\mathbb{N}$. Then, a vertex set $S \subseteq V(G)$ is called a *distance-r (σ,ρ)-dominating set*, if

- for each vertex $v \in S$ it holds that $|N^r(v) \cap S| \in \sigma$, and
- for each vertex $v \in V(G) \setminus S$ it holds that $|N^r(v) \cap S| \in \rho$.

We associate with these sets minimization, maximization, and existence problems in the natural way.

The *d-value* of a distance-r (σ,ρ) problem is a constant which will ultimately affect the runtime of the algorithm. For a set $\mu \subseteq \mathbb{N}$, the value $d(\mu)$ should be understood as the highest value in $\mathbb{N}$ we need to enumerate in order to describe μ. Hence, if μ is finite, it is simply the maximum value in μ, and if μ is co-finite, it is the maximum natural number *not* in μ (1 is added for technical reasons).

Definition 1 (d-value)**.** Let $d(\mathbb{N}) = 0$. For every non-empty finite or co-finite set $\mu \subseteq \mathbb{N}$, let $d(\mu) = 1 + \min(\max\{x \mid x \in \mu\}, \max\{x \mid x \in \mathbb{N} \setminus \mu\})$.

For a given distance-r (σ,ρ) problem $\Pi_{\sigma,\rho}$, its d-value is defined as $d(\Pi_{\sigma,\rho}) := \max\{d(\sigma), d(\rho)\}$, see column d in Table 1.

2.3 Mim-width and Applications

Maximum induced matching width, or *mim-width* for short, was introduced in the Ph. D. thesis of Vatshelle [31], used implicitly by Belmonte and Vatshelle [3], and is a structural graph parameter defined over *decomposition trees* (sometimes called *branch decompositions*), similar to graph parameters such as *rank-width* and *module-width*. Decomposition trees naturally appear in divide and conquer style algorithms where one recursively partitions the pieces of a problem into two parts. When the algorithm is at the point where it combines solutions of its subproblems to form a full solution, the structure of the cuts are (unsurprisingly) important to the runtime; this is especially true for dynamic programming when one needs to store multiple partial solutions at each intermediate node. We will briefly introduce the necessary machinery here, but for a more comprehensive introduction we refer the reader to [31].

A graph of maximum degree at most 3 is called *subcubic*. A *decomposition tree* for a graph G is a pair $(T, \mathcal{L})$ where T is a subcubic tree and $\mathcal{L} : V(G) \to L(T)$ is a bijection between the vertices of G and the leaves of T. Each edge $e \in E(T)$ naturally represents a cut of G, i. e. a 2-partition of $V(G)$: Let T_1^e and T_2^e denote the two connected components of $T - e$. For $i \in [2]$, let A_i^e denote the set of vertices that are mapped to leaves in T_i^e via $\mathcal{L}$. Then, (A_1^e, A_2^e) is a cut of G which in the following we refer to as the *cut associated with* e. One way to measure the complexity of a cut is by considering the maximum size of an induced matching in the bipartite subgraph of G that it induces. For a vertex set $A \subseteq V(G)$, we let $\mathrm{mim}_G(A)$ denote the maximum size of an induced matching in $G[A, V(G) \setminus A]$, the bipartite subgraph of G induced by $(A, V(G) \setminus A)$. Note that mim_G is symmetric, i.e. $\mathrm{mim}_G(A) = \mathrm{mim}_G(V(G) \setminus A)$.

Definition 2 (mim-width)**.** Let G be a graph, and let $(T, \mathcal{L})$ be a decomposition tree for G. The *mim-width* of $(T, \mathcal{L})$ is defined as

$$\mathrm{mimw}_G(T, \mathcal{L}) \coloneqq \max_{e \in E(T)} \mathrm{mim}_G(A_1^e),$$

where for an edge $e \in E(T)$, (A_1^e, A_2^e) denotes the cut associated with e. The *mim-width of the graph* G, denoted $\mathrm{mimw}(G)$, is the minimum value of $\mathrm{mimw}_G(T, \mathcal{L})$ over all possible decomposition trees $(T, \mathcal{L})$. The *linear mim-width of the graph* G is the minimum value of $\mathrm{mimw}_G(T, \mathcal{L})$ over all possible decomposition trees $(T, \mathcal{L})$ where T is a caterpillar.

Note that the definition of a decomposition tree $(T, \mathcal{L})$ allows T to have nodes of degree two. Suppose there is a node $t \in V(T)$ of degree two with incident edges e_1 and e_2. Then, up to renaming the sets, we have that $A_1^{e_1} = A_1^{e_2}$ and $A_2^{e_1} = A_2^{e_2}$, where for $i \in [2]$, $(A_1^{e_i}, A_2^{e_i})$ is the cut associated with e_i. Let t_i denote the endpoint of e_i other than t. The observation we just made implies that the decomposition tree $(T', \mathcal{L})$, where T' is obtained from T by removing t and making t_1 and t_2 adjacent, is equivalent to $(T, \mathcal{L})$ both in terms of their mim-width, as well as their structural properties. To that end, for ease of exposition, we will assume in algorithmic applications that a decomposition tree does not have degree two nodes, while when proving bounds on the mim-width of some decomposition tree, we may allow them to have degree two nodes.

Given a decomposition tree, one can subdivide an arbitrary edge and let the newly created vertex be the root of T, in the following denoted by r. For two nodes $t, t' \in V(T)$, we say that t' is a *descendant* of t if t lies on the path from r to t' in T. For $t \in V(T)$, we denote by V_t the set of vertices that are mapped to a leaf that is a descendant of t, i.e. $V_t \coloneqq \{v \in V(G) \mid \mathcal{L}(v) = t'$ where t' is a leaf descendant of t in $T\}$. We let $\overline{V_t} \coloneqq V(G) \setminus V_t$ and $G_t \coloneqq G[V_t]$.

In previous work, Bui-Xuan et al. [7] and Belmonte and Vatshelle [3] showed that all (σ, ρ) problems can be solved in time $n^{\mathcal{O}(w)}$ where w denotes the mim-width of a decomposition tree that is provided as part of the input. As we state the runtime bounds more tightly than what can be read from the statements in the original works, we provide a sketch of the proof of the following result due to [3, 7], mainly focusing on aspects that affect the runtime rather than the correctness of the algorithms.

Proposition 3 ([3, 7]). *There is an algorithm that given a graph G and a decomposition tree $(T, \mathcal{L})$ of G with $w := mimw_G(T, \mathcal{L})$ solves each (σ, ρ) problem Π with $d := d(\Pi)$*

(i) in time $\mathcal{O}(n^{4+2d\cdot w})$, if T is a caterpillar, and

(ii) in time $\mathcal{O}(n^{4+3d\cdot w})$, otherwise.

Sketch of the Proof. For a positive integer d, and a set $A \subseteq V(G)$, we call two subsets $X \subseteq A$ and $Y \subseteq A$ *d-neighbor equivalent w.r.t. A*, and we write $X \equiv^d_A Y$, if

$$\forall v \in V(G) \setminus A\colon \min(d, |X \cap N(v)|) = \min(d, |Y \cap N(v)|).$$

We denote by $\mathrm{nec}(\equiv^d_A)$ the number of equivalence classes of $\equiv^d_A$, and for $X \subseteq A$, by $\mathrm{rep}^d_A(X)$ an equivalence class representative for X.

The crucial insight that makes the dynamic programming algorithm work within the claimed runtime bound is: when tabulating the partial solutions with respect to a node $t \in V(T)$, it suffices to store one optimum partial solution for each pair $(R_t, R_{\overline{t}})$, where R_t is an equivalence class representative for $\equiv^d_{V_t}$ and $R_{\overline{t}}$ is an equivalence class representative for $\equiv^d_{\overline{V_t}}$. In this way, an upper bound on $\mathrm{nec}(\equiv^d_A)$ gives an upper bound on the number of table entries to be considered.

Now, let ℓ be a leaf of T and $v \in V(G)$ be such that $\mathcal{L}(v) = \ell$. Then, there are two equivalence classes for $\equiv^d_{\{v\}}$, and $d+1$ equivalence classes for $\equiv^d_{V(G)\setminus\{v\}}$, so we only have $\mathcal{O}(d)$ partial solutions to consider. For an internal node $t \in V(T)$ with children a and b, we can compute the necessary partial solutions in the following way. For each triple $(R_a, R_b, R_{\overline{t}})$ of an equivalence class representative R_a of $\equiv^d_{V_a}$, an equivalence class representative R_b of $\equiv^d_{V_b}$, and an equivalence class representative $R_{\overline{t}}$ of $\equiv^d_{\overline{V_t}}$, compute $R_t = \mathrm{rep}^d_{V_t}(R_a \cup R_b)$, $R_{\overline{a}} = \mathrm{rep}^d_{\overline{V_a}}(R_b \cup R_{\overline{t}})$, and $R_{\overline{b}} = \mathrm{rep}^d_{\overline{V_b}}(R_a \cup R_{\overline{t}})$, and update the table entry for $(R_t, R_{\overline{t}})$ according to the partial solution obtained from combining the partial solution stored for $(R_a, R_{\overline{a}})$ with the one stored for $(R_b, R_{\overline{b}})$.

Let $\mathrm{nec}_d(T, \mathcal{L}) = \max_{t\in V(T)} \max\{\mathrm{nec}(\equiv^d_{V_t}), \mathrm{nec}(\equiv^d_{\overline{V_t}})\}$. We argue that for each internal node $t \in V(T)$, all table entries can be computed in time $\mathcal{O}(\mathrm{nec}_d(T, \mathcal{L})^3 \cdot n^3)$. It can be shown [7, Lemma 1] that for a set $A \subseteq V(G)$, a set of canonical equivalence class representatives of $\equiv^d_A$ can be enumerated in time $\mathcal{O}(\mathrm{nec}(\equiv^d_A) \cdot \log(\mathrm{nec}(\equiv^d_A)) \cdot n^2)$, and given a set $X \subseteq A$, one can find a pointer to $\mathrm{rep}^d_A(X)$ in time $\mathcal{O}(\log(\mathrm{nec}(\equiv^d_A)) \cdot |X| \cdot n)$.

The former computation is invoked once, taking time at most $\mathcal{O}(\mathrm{nec}_d(T, \mathcal{L}) \cdot \log(\mathrm{nec}_d(T, \mathcal{L})) \cdot n^2)$. Next, there are at most $\mathrm{nec}_d(T, \mathcal{L})^3$ triples of equivalence class representatives to consider, and the remaining computations take time at most $\mathcal{O}(n^3)$, applying the latter of the two aforementioned tasks (finding a pointer to an equivalence class representative), and noting that each representative is of size at most $\mathcal{O}(n)$, and that $\log(\mathrm{nec}_d(T, \mathcal{L})) \le \mathcal{O}(n)$. Therefore, we compute all table entries for $t \in V(T)$ in time at most $\mathcal{O}(\mathrm{nec}_d(T, \mathcal{L})^3 \cdot n^3)$. As $|V(T)| = \mathcal{O}(n)$, and the computation for a leaf node takes constant time, the whole algorithm takes time at most $\mathcal{O}(\mathrm{nec}_d(T, \mathcal{L})^3 \cdot n^4)$.

In [3, Lemma 2] it is shown that $\operatorname{nec}(\equiv^d_A) \leq n^{d\cdot\operatorname{mim}(A)}$. Hence the dynamic programming algorithms that we sketched above run in time $\mathcal{O}(n^{4+3d\cdot w})$, where $w = \operatorname{mimw}(T, \mathcal{L})$, proving item (ii). For item (i), we observe that for an internal node $t \in V(T)$ with children a and b such that b is a leaf node, the number of triples to consider reduces to $\mathcal{O}(\operatorname{nec}_d(T, \mathcal{L})^2)$, as there are only $\mathcal{O}(1)$ many equivalence class representatives to consider for $\equiv^d_{V_b}$. If T is a caterpillar, then each internal node of T is of that shape, implying that in this case, the algorithms run in time $\mathcal{O}(n^{4+2d\cdot w})$. $\square$

3 Mim-width on Graph Powers

In this section we discuss the structural results of this work, regarding the mim-width of graph powers. These are formally defined as follows.

Definition 4 (The r-th Power of a Graph)**.** Let r be a positive integer and let $G = (V, E)$ be a graph. The r-th power of G, denoted G^r, is the graph obtained from G by adding, for each pair of distinct non-adjacent vertices $u, v \in V(G)$ with $\textsc{dist}_G(u, v) \leq r$, the edge uv.

We show in Section 3.1 that taking an (arbitrarily large) power of a graph only increases its mim-width by a factor of at most two. In Section 3.2 we prove results concerning powers of graphs of bounded tree-width and clique-width.

3.1 Arbitrary Graphs

Theorem 5. *Let r be a positive integer and let G be a graph. Then,* $\operatorname{mimw}(G^r) \leq 2 \cdot \operatorname{mimw}(G)$*. Moreover, any decomposition tree of G of mim-width w has mim-width at most $2w$ for G^r.*

Proof. Assume that there is a decomposition tree of mim-width w for the graph G. We show that the same decomposition tree has mim-width at most $2w$ for G^r.

We consider a cut $(A, \overline{A})$ associated with some edge of the decomposition tree. Let M be a maximum induced matching across the cut for G^r. To prove our claim, it suffices to construct an induced matching across the cut M' in G such that $|M'| \geq \frac{|M|}{2}$.

We begin by noticing that for an edge $uv \in M$, the distance between u and v is at most r in G. For each such edge $uv \in M$, we let P_{uv} denote some shortest path between u and v in G (including the endpoints u and v).

Claim 5.1. Let $uv, wx \in M$ be two distinct edges of the matching. Then P_{uv} and P_{wx} are vertex disjoint.

Proof. We may assume that $u, w \in A$ and $v, x \in \overline{A}$. Now assume for the sake of contradiction there exists a vertex $y \in V(P_{uv}) \cap V(P_{wx})$. Because both paths have length at most r, we have that $\textsc{dist}_G(u, y) + \textsc{dist}_G(y, v) \leq r$, and $\textsc{dist}_G(w, y) + \textsc{dist}_G(y, x) \leq r$. Adding these together, we get

$$\textsc{dist}_G(u, y) + \textsc{dist}_G(y, v) + \textsc{dist}_G(w, y) + \textsc{dist}_G(y, x) \leq 2r.$$

Since uv and wx are both in M, there cannot exist edges ux and wv in G^r. Hence, their distance in G is strictly greater than r, i.e. $\textsc{dist}_G(u, y) + \textsc{dist}_G(y, x) \geq \textsc{dist}_G(u, x) > r$, and $\textsc{dist}_G(w, y) + \textsc{dist}_G(y, v) > r$. Putting these together, we obtain our contradiction:

$$\textsc{dist}_G(u, y) + \textsc{dist}_G(y, x) + \textsc{dist}_G(w, y) + \textsc{dist}_G(y, v) > 2r$$

This concludes the proof of the claim. ⌟

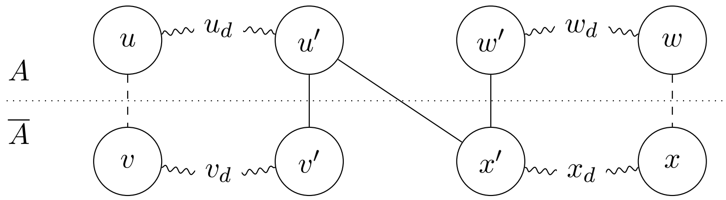


Figure 1: Structure of two paths P_{uv} and P_{wx} when the edge $u'x'$ exists in G. Dashed edges appear in G^r, solid edges appear in G, squiggly lines are (shortest) paths existing in G (possibly of length 0, and possibly crossing back and forth across the cut).

Because for each $uv \in M$, one endpoint of P_{uv} lies in A and the other one lies in $\overline{A}$, there must exist at least one point at which the path crosses from A to $\overline{A}$. For each $uv \in M$ with $u \in A$ and $v \in \overline{A}$, we let $u'v' \in E(P_{uv})$ denote an edge in G such that $u' \in A$ and $v' \in \overline{A}$.

We plan to construct our matching M' by picking a subset of such edges. However, we cannot simply take all of them, since some pairs may be incompatible in the sense that they will not form an induced matching across the cut $(A, \overline{A})$. We examine the structures that arise when two such edges $u'v'$ and $w'x'$ are incompatible, and cannot both be included in the same induced matching across the cut. For easier readability, we let α_d be a shorthand notation for $\textsc{dist}_G(\alpha, \alpha')$ for $\alpha \in \{u, v, w, x\}$.

Claim 5.2. Let $uv, wx \in M$ with $\{u, w\} \subseteq A$ and $\{v, x\} \subseteq \overline{A}$ be two distinct edges of M and let $u'v'$ and $w'x'$ be edges on the shortest paths as defined above. If there is an edge $u'x' \in E(G)$, then all of the following hold. See Figure 1.

(a) $u_d + x_d = r$

(b) $u_d + v_d = w_d + x_d = r - 1$

(c) $w_d = u_d - 1$

Proof. (a) Since ux is not an edge in G^r, the distance between u and x must be at least $r + 1$ in G, and so $u_d + x_d$ must be at least r. It remains to show that $u_d + x_d \leq r$ for equality to hold. Similarily to the proof of Claim 5.1, we know that P_{uv} and P_{wx} both are of length at most r. We get

$$u_d + v_d + w_d + x_d \leq 2r - 2 \tag{1}$$

The '-2' at the end is because we do not include the length contributed by edges $u'v'$ and $w'x'$ in our sum. Now assume for the sake of contradiction that $u_d + x_d \geq r + 1$. Then we get that

$$v_d + w_d \leq 2r - 2 - r - 1 = r - 3$$

Because $\textsc{dist}_G(v', w') \leq 3$ (follow the edges $u'v' \to u'x' \to w'x'$), this implies that $\textsc{dist}_G(v, w) \leq r$, and the edge vw would hence exist in G^r. This contradicts that uv and wx were both in the same induced matching M.

(b) Assume for the sake of contradiction that $u_d + v_d \leq r - 2$. Then, rather than Equation 1, we get the following bound

$$u_d + v_d + w_d + x_d \leq 2r - 3$$

By (a) we know that $u_d + x_d = r$, so by a similar argument as above we get that $v_d + w_d \leq r - 3$, obtaining a contradiction. An analogous argument holds for $w_d + x_d$.

(c) This follows immediately by substituting (a) into (b). ⌟

We will now construct our induced matching M'. (Recall for the following arguments that $u \in A$ and $v \notin A$.) We construct two candidates for M', and we will pick the biggest one. First, we construct M'_0 by including $u'v'$ for each edge $uv \in M$ where $\textsc{dist}_G(u, u')$ is even. Symetrically, M'_1 is constructed by including $u'v'$ if $\textsc{dist}_G(u, u')$ is odd. Clearly, at least one of M'_0 and M'_1 contains at least $\frac{|M|}{2}$ edges. It remains to show that M' indeed forms an induced matching across the cut $(A, \overline{A})$ in G.

Consider two distinct edges $u'v'$ and $w'x'$ from M'. By Claim 5.1, the corresponding paths P_{uv} and P_{wx} are vertex disjoint. If there is an edge violating that $u'v'$ and $w'x'$ are both in the same induced matching, it must be either $u'x'$ or $v'w'$. Without loss of generality we may assume it is an edge of the type $u'x'$. By Claim 5.2(c), we then have that the parities of $\textsc{dist}_G(u, u')$ and $\textsc{dist}_G(w, w')$ are different. However, by the construction of M', this is not possible. This concludes the proof. $\square$

3.2 Graphs of Bounded Tree-Width or Clique-Width

In [31, Section 4.2], it is shown that any graph of clique-width w or tree-width $w - 1$ has mim-width at most w. Theorem 5 hence implies that any power of a graph of clique-width w or tree-width $w - 1$ has mim-width at most $2w$. In this section we give tighter bounds on the mim-width of powers of graphs of bounded clique-width and powers of graphs of bounded tree-width.

In particular, we show that r-th powers of graphs of tree-width $w - 1$, path-width w, or clique-width w all have mim-width at most w. We begin by proving the bound for graphs of bounded tree-width with the following lemma capturing the essential property used in the proof.

Lemma 6. *Let r and w be positive integers and let (A, B, C) be a vertex partition of graph G such that there are no edges between A and C and B has size w. Let $H \coloneqq G^r$. Then,* $\mathrm{mim}_H(A \cup B) \leq w$.

Proof. Let $B \coloneqq \{b_1, b_2, \ldots, b_w\}$. It is clear that for $v \in A \cup B$ and $z \in C$, $\textsc{dist}_G(v, z) \leq r$ if and only if there exists $i \in \{1, 2, \ldots, w\}$ such that $\textsc{dist}_G(v, b_i) + \textsc{dist}_G(z, b_i) \leq r$.

Suppose for a contradiction that $\mathrm{mim}_H(A \cup B) > w$. Let $\{y_1z_1, y_2z_2, \ldots, y_tz_t\}$ be an induced matching of size $t \geq w + 1$ in $H[A \cup B, C]$. There are distinct integers $t_1, t_2 \in \{1, 2, \ldots, t\}$ and an integer $j \in \{1, 2, \ldots, w\}$ such that

$$\textsc{dist}_G(y_{t_1}, b_j) + \textsc{dist}_G(z_{t_1}, b_j) \leq r \text{ and } \textsc{dist}_G(y_{t_2}, b_j) + \textsc{dist}_G(z_{t_2}, b_j) \leq r.$$

Then we have either $\textsc{dist}_G(y_{t_1}, b_j) + \textsc{dist}_G(z_{t_2}, b_j) \leq r$ or $\textsc{dist}_G(y_{t_2}, b_j) + \textsc{dist}_G(z_{t_1}, b_j) \leq r$, which contradicts the assumption that $y_{t_1}z_{t_2}$ and $y_{t_2}z_{t_1}$ are not edges in H. We conclude that $\mathrm{mim}_H(A \cup B) \leq w$. $\square$

Definition 7. A *tree decomposition* of a graph G is a pair $(T, \mathcal{B})$ consisting of a tree T and a family $\mathcal{B} = \{B_t\}_{t \in V(T)}$ of sets $B_t \subseteq V(G)$, called *bags*, satisfying the following three conditions:

(i) $V(G) = \bigcup_{t \in V(T)} B_t$,

(ii) for every edge uv of G, there exists a node t of T such that $u, v \in B_t$, and

(iii) for $t_1, t_2, t_3 \in V(T)$, $B_{t_1} \cap B_{t_3} \subseteq B_{t_2}$ whenever t_2 is on the path from t_1 to t_3 in T.

The *width* of a tree decomposition $(T, \mathcal{B})$ is $\max\{|B_t| - 1 : t \in V(T)\}$. The *tree-width* of G is the minimum width over all tree decompositions of G. A tree decomposition $(T, \mathcal{B} = \{B_t\}_{t \in V(T)})$ is a *nice tree decomposition* with root node $\mathfrak{r} \in V(T)$ if T is a rooted tree with root node $\mathfrak{r}$, and every node t of T is one of the following:

(1) A *leaf node*, i.e. t is a leaf of T and $B_t = \emptyset$.

(2) An *introduce node*, i.e. t has exactly one child t' and $B_t = B_{t'} \cup \{v\}$ for some $v \in V(G) \setminus B_{t'}$.

(3) A *forget node*, i.e. t has exactly one child t' and $B_t = B_{t'} \setminus \{v\}$ for some $v \in B_{t'}$.

(4) A *join node*, i.e. t has exactly two children t_1 and t_2, and $B_t = B_{t_1} = B_{t_2}$.

Theorem 8. *Let r and w be positive integers and G be a graph that admits a nice tree decomposition of width w, all of whose join bags are of size at most w. Then the r-th power of G has mim-width at most w. Furthermore, given such a nice tree decomposition, we can output a decomposition tree of mim-width at most w in polynomial time.*

Proof. Let $H := G^r$, and let $(T, \mathcal{B} = \{B_t\}_{t \in V(T)})$ be a nice tree decomposition of G of width w, all of whose join bags have size at most w, with root node $\mathfrak{r}$. We may assume that $B_{\mathfrak{r}} = \emptyset$ and subsequently that $\mathfrak{r}$ is a forget or a join node. (Otherwise, we add a path of forget nodes on top of $\mathfrak{r}$ and make the last node the new root of T.)

We obtain a decomposition tree $(T', \mathcal{L})$ as follows:

- Let T'' be the tree obtained from T by, for each forget node $t \in V(T)$ forgetting a vertex v, attaching a leaf node ℓ_v to t, and assigning $\mathcal{L}(v) := \ell_v$.

- We obtain T' from T'' by deleting degree 1 nodes that are not assigned by $\mathcal{L}$.

Since $\mathfrak{r}$ is either a forget node or a join node in $(T, \mathcal{B})$, $\mathfrak{r}$ has not been removed in the second step, so $\mathfrak{r} \in V(T')$, and $\mathfrak{r}$ has degree 2 in T'. Furthermore, since for each vertex $v \in V(G)$, there is a unique forget node forgetting v in $(T, \mathcal{B})$, and all leaves that are not assigned by $\mathcal{L}$ have been removed, the map $\mathcal{L}$ constructed above is a bijection. Thus, $(T', \mathcal{L})$ is a (rooted) decomposition tree with root node $\mathfrak{r}$.

We consider a cut $(V_t, \overline{V_t})$ for some node $t \in V(T')$ in the rooted decomposition tree. If t is a leaf node, then $\text{mim}_H(V_t) \le 1$. Assume t is an internal node, then t also appears in $(T, \mathcal{B})$. Note that V_t consists precisely of all vertices that have been forgotten below t in $(T, \mathcal{B})$. We argue that we can find a set of at most w vertices $S \subseteq \overline{V_t}$ such that S separates V_t from $\overline{V_t} \setminus S$ which by Lemma 6 will yield the claim.

Let $S := N(V_t) \cap \overline{V_t}$ and consider a vertex $x \in S$. By definition, x is a neighbor of some vertex y that has been forgotten below t. By (ii) of the definition of a tree decomposition there is a node, say t', such that $B_{t'}$ contains both x and y. Since y has been forgotten below t, t' is below t. As $x \in \overline{V_t}$, there is also a node above t, say t'', such that $B_{t''}$ contains x (e.g. the bag below the one that forgets x). Since t lies on the path between t' and t'' in T, we have by property (iii) of the definition of a tree decomposition that $x \in B_t$. We have that $S \subseteq B_t$.

Furthermore, S separates V_t from $\overline{V_t} \setminus S$. If t is a forget node, then by definition $|B_t| \le w$ and hence $|S| \le w$. If t is a join node, then by assumption $|B_t| \le w$ and hence $|S| \le w$. If t is an introduce node introducing a vertex $u \in V(G)$, then u cannot have any neighbor in V_t, since all vertices in V_t have been forgotten below t. Hence, $S \subseteq B_t \setminus \{u\}$ and we can conclude that $|S| \le w$. ∎

It is well-known (see e.g. [21]) that any tree decomposition can be transformed in polynomial time to a nice tree decomposition of the same width. As in a tree decomposition of width $w-1$, all bags (in particular bags at join nodes) have size at most w, the previous theorem implies the following. Note that the bound for pathwidth is slightly tighter than the one for tree-width, as in a path decomposition there are no join nodes.

Corollary 9. *Let r and w be positive integers and let G be a graph of tree-width $w-1$ (path-width w). Then the r-th power of G has mim-width at most w and given a tree decomposition (path decomposition) of G of width $w-1$ (w), one can compute a decomposition tree of mim-width w in polynomial time.*

The following notions are of importance in the field of phylogenetic studies, i.e. the reconstruction of ancestral relations in biology, see e.g. [8]. A graph G is a *leaf power* if there exists a threshold r and a tree T, called a leaf root, whose leaf set is $V(G)$ such that $uv \in E$ if and only if the distance between u and v in T is at most r. Similarly, G is called a *min-leaf power* if $uv \in E$ if and only if the distance between u and v in T is more than r. Thus, G is a leaf power if and only if its complement is a min-leaf power. It is easy to see that trees admit nice tree decompositions all of whose join bags have size 1 and since every leaf power graph is an induced subgraph of a power of some tree, it has mim-width at most 1 by Theorem 8. Together with [31, Lemma 3.7.3], stating that the mim-width of a graph is 1 if and only if the mim-width of its complement is 1, we have the following result.

Corollary 10. *The leaf powers and min-leaf powers have mim-width at most* 1 *and given a leaf root, we can compute in polynomial time a decomposition tree witnessing this.*

Next, we consider powers of graphs of bounded clique-width. A graph is *w-labeled* if there is a labeling function $f : V(G) \to [w]$, and we call $f(v)$ the *label* of v. For a w-labeled graph G, we call the set of all vertices with label i the *label class* i of G. The following can be thought as a generalization of Lemma 6.

Lemma 11. *Let r and w be positive integers and let (A, B) be a vertex partition of graph G such that $G[A]$ is w-labeled and for every pair of vertices x, y in the same label class of $G[A]$, x and y have the same neighborhood in B. Let $H := G^r$. Then,* $\mathrm{mim}_H(A) \leq w$.

Proof. Suppose for contradiction that $\mathrm{mim}_H(A) > w$. Let $\{y_1z_1, y_2z_2, \ldots, y_tz_t\}$ be an induced matching of size at least $w+1$ in $H[A, B]$. For $i \in \{1, 2, \ldots, t\}$, there is a path P_i of length at most r from y_i to z_i in G. Let $A^* \subseteq A$ be the set of vertices in A that have a neighbor in B. Let Q_i be the subpath of P_i from the last vertex in A^* to z_i, and q_i be the endpoint of Q_i different from z_i, and let R_i be the subpath of P_i from y_i to q_i. Let a_i be the length of R_i and b_i be the length of Q_i. By construction, $a_i + b_i \leq r$.

(For an illustration of the following argument, see Figure 2.) Since $t \geq w+1$, there are two integers $t_1, t_2 \in \{1, 2, \ldots, t\}$ such that q_{t_1} and q_{t_2} are contained in the same label class of $G[A]$. Since $a_{t_1} + b_{t_1} \leq r$ and $a_{t_2} + b_{t_2} \leq r$, we either have that $a_{t_1} + b_{t_2} \leq r$ or $a_{t_2} + b_{t_1} \leq r$.

Assume $a_{t_1} + b_{t_2} \leq r$. In this case, we show that the distance from y_{t_1} to z_{t_2} in G is at most r, which contradicts the assumption that $y_{t_1}z_{t_2}$ is not an edge of H. Note that two vertices in a label class of $G[A]$ have the same neighborhood in B. Let x denote the vertex on Q_{t_2} that is adjacent to q_{t_2}. Then, as q_{t_1} and q_{t_2} are in the same label class of $G[A]$, we have that q_{t_1} is also adjacent to x. Therefore, $G[V(R_{t_1}) \cup (V(Q_{t_2}) \setminus \{q_{t_2}\})]$ contains a path of length at most r from y_{t_1} to z_{t_2}.

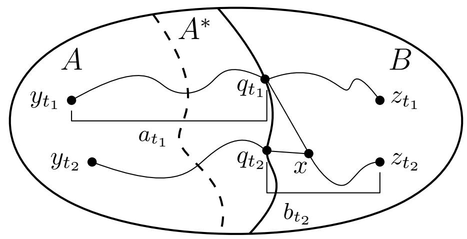


Figure 2: Illustration of the situation in the proof of Lemma 11.

Analogously we can show that if $a_{t_2} + b_{t_1} \leq r$, then $G[V(R_{t_2}) \cup (V(Q_{t_1}) \setminus \{q_{t_1}\})]$ contains a path of length at most r from y_{t_2} to z_{t_1}. But this is a contradiction and we conclude that $\text{mim}_H(A) \leq w$. $\square$

Definition 12. The *clique-width* of a graph G is the minimum number of labels needed to construct G using the following four operations:

(1) Creation of a new vertex v with label i (denoted by $i(v)$).

(2) Disjoint union of two labeled graphs G and H (denoted by $G \oplus H$).

(3) Joining by an edge each vertex with label i to each vertex with label j ($i \neq j$, denoted by $\eta_{i,j}$).

(4) Renaming label i to j (denoted by $\rho_{i \to j}$).

A string of operations given in the previous definition is called a *clique-width w-expression* or shortly a *w-expression* if it uses at most w distinct labels. We can represent this expression as a tree-structure and such trees are known as *syntactic trees* associated with w-expressions. An easy observation is that for a node t in a syntactic tree associated with a w-expression, and the vertex set V_t consisting of vertices introduced in some descendants of t, $G[V_t]$ is a w-labeled graph where two vertices in the same label class has the same neighborhood in $V(G) \setminus V_t$.

Theorem 13. *Let r and w be positive integers and G be a graph of clique-width w. Then the r-th power of G has mim-width at most w. Furthermore, given a clique-width w-expression, we can output a decomposition tree of mim-width at most w in polynomial time.*

Proof. Let H be the r-th power of G and let ϕ be the given clique-width w-expression defining G, and T be the syntactic tree of ϕ with root node $\mathfrak{r}$. We can assume that G contains at least two vertices which implies that T has at least one join node. Let $\mathfrak{r}'$ be the join node in T with minimum $\textsc{dist}_T(\mathfrak{r}, \mathfrak{r}')$. Note that if $\mathfrak{r}$ itself is a join node, then $\mathfrak{r}' = \mathfrak{r}$. Note also that for every vertex v of G, there is a node of T creating v, see the first operation in the definition of clique-width. In the following, we call such a node an *introduce node*. We obtain a decomposition tree $(T', \mathcal{L})$ as follows:

- We obtain T' from T as follows: If $\mathfrak{r} \neq \mathfrak{r}'$, we first remove all vertices in the path from $\mathfrak{r}'$ to $\mathfrak{r}$ in T other than $\mathfrak{r}'$. We fix $\mathfrak{r}'$ to be the root node of T'.

- For each introduce node ℓ_v introducing a vertex v, we assign $\mathcal{L}(v) := \ell_v$.

For the first step, note that if the root $\mathfrak{r}$ of T is not a join node, then it must have degree one in T. This implies that T' is connected, and that each introduce node is a descendant of $\mathfrak{r}'$. Consider a cut $(V_t, \overline{V_t})$ for some $t \in V(T')$, where V_t is the set of vertices that are introduced below t in T, and note that by construction this is also the set of vertices of G that are mapped to leaves in the subtree of T' rooted at t. If t is a leaf node, then $\text{mim}_H(V_t) \leq 1$. Assume t is an internal node. Then t also appears in T.

We observe that $G[V_t]$ is a w-labeled graph such that for any pair of vertices x, y in the same label class, x and y have the same neighborhood in $V(G) \setminus V_t$. So we can apply Lemma 11 to conclude that we have $\text{mim}_H(V_t) \leq w$ which implies that H has mim-width at most w. □

4 Algorithmic Consequences

Using the results from the previous section, we now give algorithmic consequences for distance-r versions of (σ, ρ) problems and more generally of LCVP problems (introduced below). In particular, the results in this section follow from Theorem 5 which states that taking an arbitrary power of a graph never increases its mim-width by more than a factor of two. The second ingredient is the following simple observation about neighborhoods of r-th powers of graphs.

Observation 14. For a positive integer r, a graph G and a vertex $u \in V(G)$, the r-neighborhood of u is equal to the neighborhood of u in G^r, i.e. $N^r_G(u) = N_{G^r}(u)$.

The observation above shows that solving a distance-r (σ, ρ) problem on G is the same as solving the same standard distance-1 variation of the problem on G^r. Hence, we may reduce our problem to the standard version by simply computing the graph power. Combining Theorem 5 with the algorithms provided in Proposition 3, we have the following consequence.

Corollary 15. *There is an algorithm that for all $r \in \mathbb{N}$, given a graph G and a decomposition tree $(T, \mathcal{L})$ of G with $w := mimw_G(T, \mathcal{L})$ solves each distance-r (σ, ρ) problem Π with $d := d(\Pi)$*

(i) in time $\mathcal{O}(n^{4+4d\cdot w})$, if T is a caterpillar, and

(ii) in time $\mathcal{O}(n^{4+6d\cdot w})$, otherwise.

Proof. Let G be the input graph and $(T, \mathcal{L})$ the provided decomposition tree. We apply the following algorithm:

Step 1. Compute the graph G^r.

Step 2. Solve the standard (distance-1) version of the problem on G^r, providing $(T, \mathcal{L})$ as the decomposition tree.

Step 3. Return the answer of the algorithm ran in Step 2 without modification.

Computing G^r in Step 1 takes at most $\mathcal{O}(n^3)$ time using standard methods, Step 3 takes constant time. The worst time complexity is hence found in Step 2. By Theorem 5, the mim-width of $(T, \mathcal{L})$ on G^r is at most twice that of the same decomposition on G. The stated runtime then follows from Proposition 3. The correctness of this procedure follows immediately from Observation 14. □

LCVP Problems. A generalization of (σ, ρ) problems are the *locally checkable vertex partitioning* (LCVP) problems which we now discuss. A *degree constraint matrix* D is a $q \times q$ matrix where each entry is a finite or co-finite subset of $\mathbb{N}$. For a graph G and a partition of its vertices $\mathcal{V} = \{V_1, V_2, \ldots V_q\}$ (where some parts of the partition may possibly be empty), we say that it is a D-partition if and only if, for each $i, j \in [q]$ and each vertex $v \in V_i$, it holds that $|N(v) \cap V_j| \in D[i, j]$.

For instance, let D_3 be the 3×3 matrix whose diagonal entries are $\{0\}$ and the non-diagonal ones are $\mathbb{N}$, i. e.

$$D_3 := \begin{bmatrix} \{0\} & \mathbb{N} & \mathbb{N} \\ \mathbb{N} & \{0\} & \mathbb{N} \\ \mathbb{N} & \mathbb{N} & \{0\} \end{bmatrix}.$$

Then, a graph G admits a 3-coloring if and only if it admits a D_3-partition.

Typically, the natural algorithmic questions associated with LCVP properties are existential.[3] Interesting problems which can be phrased in such terms include the H-Covering and Graph H-Homomorphism problems where H is fixed, as well as q-coloring, Perfect Matching Cut and more. We refer to [30] for an overview.

We generalize LCVP properties to their distance-r version, by considering the ball of radius r around each vertex rather than just the immediate neighborhood.

Definition 16 (Distance-r neighborhood constraint matrix)**.** A distance-r neighborhood constraint matrix D is a $q \times q$ matrix where each entry is a finite or co-finite subset of $\mathbb{N}$. For a graph G and a partition of its vertices $\mathcal{V} = \{V_1, V_2, \dots V_q\}$ (where some parts of this partition may be empty), we say that it is a D-distance-r-partition if and only if, for each $i, j \in [q]$ and each vertex $v \in V_i$, it holds that $|N^r(v) \cap V_j| \in D[i, j]$.

We say that an algorithmic problem is a *distance-r* LCVP *problem* if the property in question can be described by a distance-r neighborhood constraint matrix. For example, the distance-r version of a problem such as q-coloring can be interpreted as an assignment of at most q colours to vertices of a graph such that no two vertices are assigned the same colour if they are at distance r or closer.

For a given distance-r LCVP problem Π, its d-value $d(\Pi)$ is the maximum d-value over all the sets in the corresponding neighborhood constraint matrix.

As in the case of (σ, ρ) problems, combining Theorem 5 with Observation 14 and the works [3, 7] we have the following result.

Corollary 17. *There is an algorithm that for all $r \in \mathbb{N}$, given a graph G and a decomposition tree $(T, \mathcal{L})$ of G with $w := mimw_G(T, \mathcal{L})$ solves each distance-r LCVP problem Π with $d := d(\Pi)$*

(i) in time $\mathcal{O}(n^{4+4qd\cdot w})$, if T is a caterpillar, and

(ii) in time $\mathcal{O}(n^{4+6qd\cdot w})$, otherwise.

As we state the runtime bounds in the previous corollary in a different way than in [7] where they have been proved first, we would like to note that this is justified by an argument parallel to the one we provided in the sketch of the proof of Proposition 3 presented in this work.

5 Lower Bounds

We show that several (σ, ρ) problems are W[1]-hard parameterized by linear mim-width plus solution size. Our reductions are based on two recent reductions due to Fomin, Golovach and Raymond [13]

[3] Note however that each (σ, ρ) problem can be stated as an LCVP problem via the matrix $D_{(\sigma,\rho)} = \begin{bmatrix} \sigma & \mathbb{N} \\ \rho & \mathbb{N} \end{bmatrix}$, so maximization or minimization of some block of the partition can be natural as well.

who showed that Independent Set and Dominating Set are $\mathsf{W}[1]$-hard parameterized by linear mim-width plus solution size. In fact they show hardness for the above mentioned problems on *H-graphs* (the parameter being the number of edges in H plus solution size) which we now define formally.

Definition 18 (H-Graph)**.** Let X be a set and $\mathcal{S}$ a family of subsets of X. The *intersection graph* of $\mathcal{S}$ is a graph with vertex set $\mathcal{S}$ such that $S, T \in \mathcal{S}$ are adjacent if and only if $S \cap T \neq \emptyset$. Let H be a (multi-) graph. We say that G is an *H-graph* if there is a subdivision H' of H and a family of subsets $\mathcal{M} \coloneqq \{M_v\}_{v \in V(G)}$ (called an *H-representation*) of $V(H')$ where $H'[M_v]$ is connected for all $v \in V(G)$, such that G is isomorphic to the intersection graph of $\mathcal{M}$. For a vertex $v \in V(G)$, we call M_v the *model* of v.

We make an immediate observation from the definition of H-graphs that will be useful in later proofs of this section. For a graph H, we can construct a H-representation of itself: We subdivide each edge of H once to obtain H'. Then, we create an H-representation $\mathcal{M}$ of H by adding for each vertex $v \in V(H)$ a model $M_v \coloneqq N_{H'}[v]$.

Observation 19. Any graph H is an H-graph.

All of the hardness results presented in this section are obtained via reductions to the respective problems on H-graphs, and the hardness for linear mim-width follows from the following proposition.

Proposition 20 (Theorem 2 in [13])**.** *Let G be an H-graph. Then, G has linear mim-width at most $2 \cdot ||H||$ and a corresponding decomposition tree can be computed in polynomial time given an H-representation of G.*

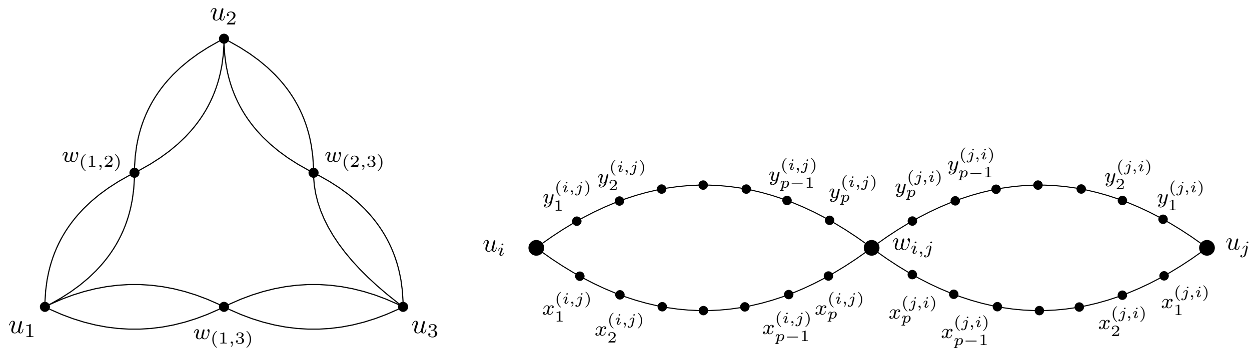
5.1 Maximization Problems

The first lower bound concerns several maximization problems that can be expressed in the (σ, ρ) framework. Recall that the (Maximum) Independent Set problem can be formulated as Max-$(\{0\}, \mathbb{N})$. The following result states that a large class of (σ, ρ) maximization problems that are related to the Independent Set problem according to their (σ, ρ) formulation are $\mathsf{W}[1]$-hard on H-graphs parameterized by $||H||$ plus solution size. These problems include Induced Matching, Dominating Induced Matching, Induced d-Regular Subgraph, and Induced Subgraph of Maximum Degree d, see Table 1 for the details.

Theorem 21. *For any fixed $d \in \mathbb{N}$ and $x \leq d+1$, the following holds. Let $\sigma^* \subseteq \mathbb{N}_{\leq d}$ with $d \in \sigma^*$. Then,* Max-$(\sigma^*, \mathbb{N}_{\geq x})$ Domination *is $\mathsf{W}[1]$-hard on H-graphs parameterized by the number of edges in H plus solution size, and the hardness holds even if an H-representation of the input graph is given.*

Proof. To prove the theorem, we provide a reduction from Multicolored Clique where given a graph G and a partition $V_1, \ldots, V_k$ of $V(G)$, the question is whether G contains a clique of size k using precisely one vertex from each V_i ($i \in [k]$). This problem is known to be $\mathsf{W}[1]$-complete [11, 25].

Let $(G, V_1, \ldots, V_k)$ be an instance of Multicolored Clique. We can assume that $k \geq 3$ and that $|V_i| = p$ for $i \in [k]$. If the second assumption does not hold, let $p \coloneqq \max_{i \in [k]} |V_i|$ and add $p - |V_i|$ isolated vertices to V_i, for each $i \in [k]$. (Note that adding isolated vertices does not change the answer to the problem.) For $i \in [k]$, we denote by $v_1^i, \ldots, v_p^i$ the vertices of V_i. We first describe the reduction of Fomin et al. [13] and then explain how to modify it to prove the theorem.



(a) Example of the graph H when $k = 3$.

(b) The subdivisions done for a pair (i, j).

Figure 3: Illustration of the construction of Fomin et al. [13].

The Construction of Fomin, Golovach and Raymond [13]. The graph H is obtained as follows, see Figure 3a.

1. Construct k nodes $u_1, \ldots, u_k$.

2. For every $1 \leq i < j \leq k$, construct a node $w_{i,j}$ and two pairs of parallel edges $u_i w_{i,j}$ and $u_j w_{i,j}$.

We then construct the subdivision H' of H by first subdividing each edge p times. We denote the subdivision nodes for 4 edges of H constructed for each pair $1 \leq i < j \leq k$ in Step 2 by $x_1^{(i,j)}, \ldots, x_p^{(i,j)}$, $y_1^{(i,j)}, \ldots, y_p^{(i,j)}$, $x_1^{(j,i)}, \ldots, x_p^{(j,i)}$, and $y_1^{(j,i)}, \ldots, y_p^{(j,i)}$. This subdivision process is depicted in Figure 3b. To simplify notation, we assume that $u_i = x_0^{(i,j)} = y_0^{(i,j)}$, $u_j = x_0^{(j,i)} = y_0^{(j,i)}$ and $w_{i,j} = x_{p+1}^{(i,j)} = y_{p+1}^{(i,j)} = x_{p+1}^{(j,i)} = y_{p+1}^{(j,i)}$.

We now construct the H-graph G'' by defining its H-representation $\mathcal{M} = \{M_v\}_{v \in V(G'')}$ where each M_v is a connected subset of $V(H')$. (Recall that G denotes the graph of the Multicolored Clique instance.)

1. For each $i \in [k]$ and $s \in [p]$, construct a vertex z_s^i with model

$$M_{z_s^i} := \bigcup_{j \in [k], j \neq i} \left(\left\{ x_0^{(i,j)}, \ldots, x_{s-1}^{(i,j)} \right\} \cup \left\{ y_0^{(i,j)}, \ldots, y_{p-s}^{(i,j)} \right\} \right).$$

2. For each edge $v_s^i v_t^j \in E(G)$ for $s, t \in [p]$ and $1 \leq i < j \leq k$, construct a vertex $r_{s,t}^{(i,j)}$ with:

$$\begin{aligned} M_{r_{s,t}^{(i,j)}} := &\left\{ x_s^{(i,j)}, \ldots, x_{p+1}^{(i,j)} \right\} \cup \left\{ y_{p-s+1}^{(i,j)}, \ldots, y_{p+1}^{(i,j)} \right\} \\ &\cup \left\{ x_t^{(j,i)}, \ldots, x_{p+1}^{(j,i)} \right\} \cup \left\{ y_{p-t+1}^{(j,i)}, \ldots, y_{p+1}^{(j,i)} \right\}. \end{aligned}$$

Throughout the following, for $i \in [k]$ and $1 \leq i < j \leq k$, we use the notation

$$Z(i) := \bigcup_{s \in [p]} \left\{ z_s^i \right\} \text{ and } R(i,j) := \bigcup_{\substack{s,t \in [p], \\ v_s^i v_t^j \in E(G)}} \left\{ r_{s,t}^{(i,j)} \right\},$$

respectively. We now observe the crucial property of G''.

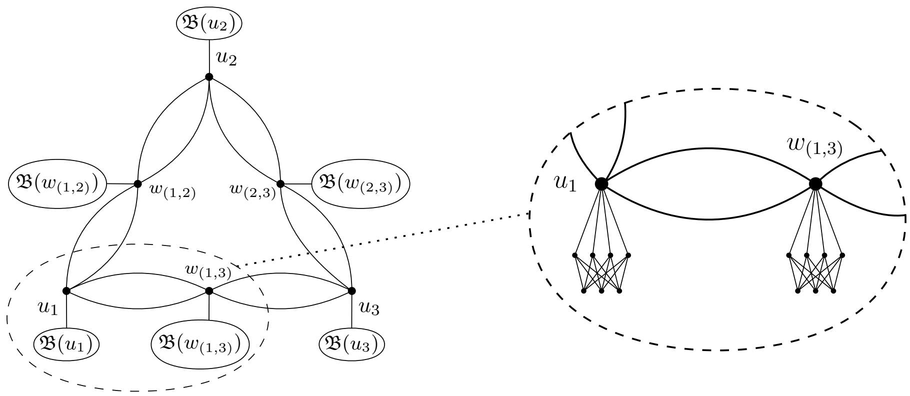


Figure 4: The graph K with respect to which the graph G' constructed in the proof of Theorem 21 is a K-graph. In this example, we have $k = 3$ and $d = 4$.

Observation 21.1 (Claim 18 in [13]). For every $1 \leq i < j \leq k$, a vertex $z_h^i \in V(G')$ (a vertex $z_h^j \in V(G')$) is *not* adjacent to a vertex $r_{s,t}^{(i,j)} \in V(G')$ corresponding to the edge $v_s^i v_t^j \in E(G)$ if and only if $h = s$ ($h = t$).

We now describe how to obtain from G'' a graph G' that will be the graph of the instance of Max-($\sigma^*, \mathbb{N}_{\geq x}$) Domination, by adding a gadget attached to each set $Z(i)$ and $R(i,j)$ (for all $1 \leq i < j \leq k$).

The New Gadget and the Construction of G'. Let X be a set of vertices of a graph. The gadget $\mathfrak{B}(X)$ is a complete bipartite graph on $2d - 1$ vertices and bipartition $(\{\beta_{1,1}, \ldots, \beta_{1,d}\}, \{\beta_{2,1}, \ldots, \beta_{2,d-1}\})$ such that for $h \in [d]$, each vertex $\beta_{1,h}$ is additionally adjacent to each vertex in X.

The graph G' is obtained from G'' by adding the gadgets $\mathfrak{B}(Z(i))$ for all $i \in [k]$ and the gadgets $\mathfrak{B}(R(i,j))$ for all $1 \leq i < j \leq k$. To prove the theorem, we require G' to be a K-graph for some graph K whose number of edges is bounded by a function of k, and possibly d, as d is fixed. We will show that G' is a K-graph for some supergraph K of H that meets this requirement.

Motivated by Observation 19, and the fact that the bipartite graph in each gadget $\mathfrak{B}(\cdot)$ has $\mathcal{O}(d^2)$ edges, we do the following to obtain K from H: For each $i \in [k]$, we add the gadget $\mathfrak{B}(\{u_i\})$, which will be used to encode $\mathfrak{B}(Z(i))$ in G'; furthermore, for each $1 \leq i < j \leq k$, we add the gadget $\mathfrak{B}(\{w_{(i,j)}\})$, which will be used to encode $\mathfrak{B}(R(i,j))$ in G'. For an illustration of K, see Figure 4. We obtain a subdivision K' of K as follows:

(K1) For all edges in $E(K) \cap E(H)$, we do the same subdivisions that were made to obtain H' from H.

(K2) For each gadget $\mathfrak{B}(\cdot)$, we perform the edge subdivisions due to Observation 19 that allow for encoding the bipartite graph in $\mathfrak{B}(\cdot)$ as a $\mathfrak{B}(\cdot)$-graph.

(K3) For each $i \in [k]$, let $\{\beta_{1,1}^i, \ldots, \beta_{1,d}^i, \beta_{2,1}^i, \ldots, \beta_{2,d-1}^i\}$ be the vertices of $\mathfrak{B}(\{u_i\})$. Then, for each $h \in [d]$, we subdivide the edge $u_i\beta_{1,h}^i$, and denote the corresponding subdivision node by $s(i,h)$.

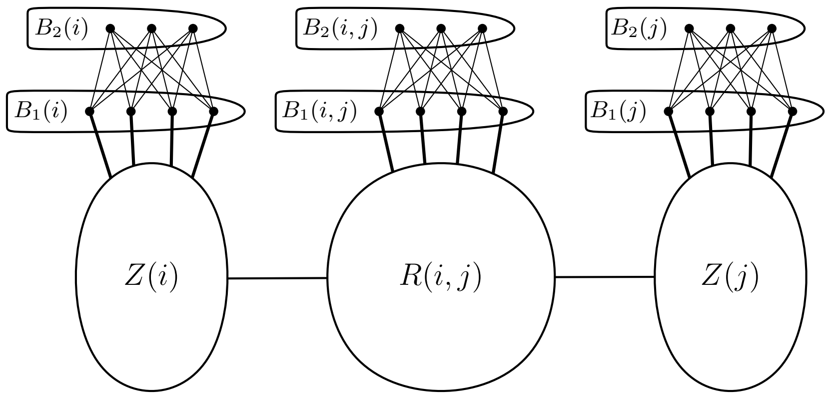


Figure 5: A part of the graph G', where $1 \leq i < j \leq k$ and $d = 4$.

(K4) Similarly, for each $1 \leq i < j \leq k$, let $\{\beta_{1,1}^{(i,j)}, \ldots, \beta_{1,d}^{(i,j)}, \beta_{2,1}^{(i,j)}, \ldots, \beta_{2,d-1}^{(i,j)}\}$ be the vertices of $\mathfrak{B}(\{w_{(i,j)}\})$. Then, for each $h \in [d]$, we subdivide the edge $u_i\beta_{1,h}^{(i,j)}$, and denote the corresponding subdivision node by $s((i,j),h)$.

We now give the K-representation of G', $\mathcal{M}' = \{M'_v\}_{v \in V(G')}$, where each M'_v is a connected subset of $V(K')$; for an illustration of G' see Figure 5.

(R1) For each vertex $v \in V(G') \cap V(G'')$, we let $M'_v := M_v$, where M_v is the model of v defined given in the construction of Fomin et al. which we described above. Note that each such vertex is either some vertex z_s^i or some vertex $r_{s,t}^{(i,j)}$ for appropriate choices for i, j, s, and t.

(R2) For each $i \in [k]$, let $\{\beta_{1,1}^i, \ldots, \beta_{1,d}^i, \beta_{2,1}^i, \ldots, \beta_{2,d-1}^i\}$ be the vertices of $\mathfrak{B}(\{u_i\})$. By the subdivisions we did in Step K2, we obtain a corresponding complete bipartite graph on vertices $\{b_{1,1}^i, \ldots, b_{1,d}^i, b_{2,1}^i, \ldots, b_{2,d-1}^i\}$. Each $b_{t,h}^i$, where $t \in [2]$ and $h \in [d]$ if $t = 1$ and $h \in [d-1]$ if $t = 2$, comes with a model from the subdivision of the complete bipartite graph of $\mathfrak{B}(\{u_i\})$, which we initially use as $M'_{b_{t,h}^i}$.

(R3) We proceed analogously to Step R2 with each $\mathfrak{B}(\{w_{(i,j)}\})$, where $1 \leq i < j \leq k$.

(R4) For each $i \in [k]$, and each $h \in [d]$, we add the node $s(i,h)$ to the models of z_s^i for all $s \in [p]$, and we add $s(i,h)$ to the model (in $\mathcal{M}'$) of $b_{1,h}^i$. (This ensures that each $b_{1,h}^i$ is complete to $Z(i)$.)

(R5) For each $1 \leq i < j \leq k$ and $s, t \in [p]$ such that $v_s^i v_t^j \in E(G)$, and each $h \in [d]$, we add the node $s((i,j),h)$ to the model of $r_{s,t}^{(i,j)}$, and we add $s((i,j),h)$ to the model (in $\mathcal{M}'$) of $b_{1,h}^{(i,j)}$. (This ensures that each $b_{1,h}^{(i,j)}$ is complete to $R(i,j)$.)

We count the size of K. For $|K|$, we observe that $|H| = k + \binom{k}{2}$ and each gadget $\mathfrak{B}(\cdot)$ has $2d-1$ nodes, and we add $k + \binom{k}{2}$ such gadgets. Hence, $|K| = 2d\left(k + \binom{k}{2}\right) = dk(k+1)$. As for $||K||$, we observe that the number of edges in H is $4 \cdot \binom{k}{2}$ and each gadget $\mathfrak{B}(\cdot)$ introduces $d(d-1) + d = d^2$ edges. Hence,

$$||K|| = 4 \cdot \binom{k}{2} + d^2\left(k + \binom{k}{2}\right) = \mathcal{O}(d^2 \cdot k^2). \tag{2}$$

We introduce some more notation. For $1 \le i < j \le k$, we let $B_1(i) \coloneqq \{b^i_{1,1}, \dots, b^i_{1,d}\}$, $B_2(i) \coloneqq \{b^i_{2,1}, \dots, b^i_{2,d-1}\}$, $B_1(i,j) \coloneqq \{b^{(i,j)}_{1,1}, \dots, b^{(i,j)}_{1,d}\}$ and $B_2(i,j) \coloneqq \{b^{(i,j)}_{2,1}, \dots, b^{(i,j)}_{2,d-1}\}$; furthermore $B(i) \coloneqq B_1(i) \cup B_2(i)$, $B(i,j) \coloneqq B_1(i,j) \cup B_2(i,j)$, and $B \coloneqq \bigcup_{i \in [k]} B(i) \cup \bigcup_{1 \le i < j \le k} B(i,j)$. Note that $|B| = (2d-1)(k + \binom{k}{2})$. We furthermore use the notation

$$Z_{+B}(i) \coloneqq Z(i) \cup B(i) \text{ and } R_{+B}(i,j) \coloneqq R(i,j) \cup B(i,j).$$

We now turn to the correctness proof of the reduction. We let $k' \coloneqq 2d \cdot (k + \binom{k}{2})$ and show that G has a multicolored clique if and only if G' has a $(\sigma^*, \mathbb{N}_{\ge x})$ set of size k'. We first prove the forward direction. Note that the following claim yields the forward direction of the correctness proof, since a $(\{d\}, \{d+1, \dots, d+k\})$ set is a $(\sigma^*, \mathbb{N}_{\ge x})$ set. (Recall that $d \in \rho^*$ and $x \le d+1$.)

Claim 21.2. If G has a multicolored clique, then G' has a $(\{d\}, \{d+1, \dots, d+k\})$ set of size $k' = 2d \cdot (k + \binom{k}{2})$ (assuming $k \ge 3$).

Proof. Let $\{v^1_{h_1}, \dots, v^k_{h_k}\}$ be the vertex set in G that induces the multicolored clique. By Observation 21.1 we can verify that

$$I \coloneqq \left\{ z^1_{h_1}, \dots, z^k_{h_k} \right\} \cup \left\{ r^{(i,j)}_{h_i, h_j} \mid 1 \le i < j \le k \right\} \tag{3}$$

is an independent set in G'. We let $S \coloneqq I \cup B$ and observe that S is a $(\{d\}, \{d+1, \dots, d+k\})$ set: By construction, there is no edge between any pair of distinct sets of $B(i)$, $B(i')$, $B(i,j)$, $B(i',j')$, for any choice of $1 \le i < j \le k$ and $1 \le i' < j' \le k$.

Consider any vertex $x \in S$ and suppose that $x \in Z_{+B}(i)$ for some $i \in [k]$. (The case when $x \in R_{+B}(i,j)$ can be argued for analogously.) If $x = z^i_{h_i}$, then x is adjacent to the d vertices $b^i_{1,1}, \dots, b^i_{1,d}$, if $x = b^i_{1,\ell}$ for some $\ell \in [d]$, then x is adjacent to $z^i_{h_i}$ and the vertices $b^i_{2,1}, \dots, b^i_{2,d-1}$ and if $x = b^i_{2,\ell'}$ for some $\ell' \in [d-1]$, then it is adjacent to the vertices $b^i_{1,1}, \dots, b^i_{1,d}$. Hence, in all cases, x has precisely d neighbors in S.

Let $y \in V(G') \setminus S$ and note that $(V(G') \setminus S) \cap B = \emptyset$. If $y \in Z(i)$ for some $i \in [k]$, then $N(y) \cap S \supseteq \{z^i_{h_i}, b^i_{1,1}, \dots, b^i_{1,d}\}$, so $|N(y) \cap S| \ge d+1$. Since the only additional neighbors of y in S are in the set $R_i \coloneqq \bigcup_{1 \le j < i} R(j,i) \cup \bigcup_{i < j \le k} R(i,j)$ and $R_i \cap S \subseteq I$, we can conclude that $|N(y) \cap (S \setminus B)| \le k-1$, since I contains precisely one vertex from each set $R(i,j)$. We have argued that $d+1 \le |N(y) \cap S| \le d+k$. If $y \in R(i,j)$ for some $1 \le i < j \le k$, we can argue as before that $|N(y) \cap S| \ge d+1$ and since all neighbors of y in $S \setminus B(i,j)$ are contained either in $Z(i)$ or $Z(j)$, we can conclude that $d+1 \le |N(y) \cap S| \le d+3 \le d+k$.

It remains to count the size of S. Clearly, $|I| = k + \binom{k}{2}$ and as observed above, $|B| = (2d-1)(k + \binom{k}{2})$, so

$$|S| = |I| + |B| = k + \binom{k}{2} + (2d-1)\left(k + \binom{k}{2}\right) = 2d\left(k + \binom{k}{2}\right) = k',$$

as claimed. ⌟

We now prove the backward direction of the correctness of the reduction. We begin by making several observations about the structure of $(\sigma^*, \mathbb{N}_{\ge x})$ sets in the graph G'.

Claim 21.3. Let $1 \le i < j \le k$.

(i) Any $(\sigma^*, \mathbb{N}_{\ge x})$ set in G' contains at most $d+1$ vertices from each $Z(i) \cup B_1(i)$ or $R(i,j) \cup B_1(i,j)$.

(ii) Any $(\sigma^*, \mathbb{N}_{\geq x})$ set contains at most $2d$ vertices from each $Z_{+B}(i)$ or $R_{+B}(i,j)$.

(iii) If a $(\sigma^*, \mathbb{N}_{\geq x})$ set S contains $2d$ vertices from some $Z_{+B}(i)$ $(R_{+B}(i,j))$, then it contains at least one vertex from $Z(i)$ $(R(i,j))$ and each such vertex in $S \cap Z(i)$ $(S \cap R(i,j))$ has at least d neighbors in $S \cap Z_{+B}(i)$ $(S \cap R_{+B}(i,j))$.

Proof. (i) We prove the claim for a set $Z(i) \cup B_1(i)$. The proof for a set $R(i,j) \cup B_1(i,j)$ works analogously. Suppose for the sake of a contradiction that there is a set $S \subseteq V(G')$ that contains at least $d+2$ vertices from some $Z(i) \cup B_1(i)$. Since $|B_1(i)| = d$, we know that S contains a vertex from $Z(i)$, say x. However, by construction, all vertices in $S \cap (Z(i) \cup B_1(i)) \setminus \{x\}$ are adjacent to x, implying that x has at least $d+1$ neighbors in S, a contradiction with the fact that S is a $(\sigma^*, \mathbb{N}_{\geq x})$ set.

(ii) follows as a direct consequence, since $Z_{+B}(i) \setminus (Z(i) \cup B_1(i)) = B_2(i)$ and $|B_2(i)| = d-1$. Similar for $R_{+B}(i,j)$.

(iii). The claim that S contains at least one vertex from $Z(i)$ is immediate since $|S \cap Z_{+B}(i)| = 2d$ and $|Z_{+B}(i) \setminus Z(i)| = |B(i)| = 2d-1$. Let $x \in S \cap Z(i)$ be such a vertex. Then, the only vertices of $Z_{+B}(i)$ that x is *not* adjacent to are the vertices $B_2(i)$. Since $|B_2(i)| = d-1$, the remaining vertices in $(S \cap Z_{+B}(i)) \setminus (B_2(i) \cup \{x\})$, of which there are at least d as we just argued, are neighbors of x. Similar for $R_{+B}(i,j)$. ⌟

Equipped with the previous claim, we can now finish the correctness proof of the reduction.

Claim 21.4. If G' contains a $(\sigma^*, \mathbb{N}_{\geq x})$ set S of size $k' = 2d(k + \binom{k}{2})$, then G contains a multicolored clique.

Proof. Let S be a $(\sigma^*, \mathbb{N}_{\geq x})$ set of size k' in G'. By Claim 21.3(ii), we can conclude that S contains precisely $2d$ vertices from each $Z_{+B}(i)$ and each $R_{+B}(i,j)$ (where $1 \leq i < j \leq k$). Consider any pair i, j with $1 \leq i < j \leq k$. By Claim 21.3(iii) we know that there are vertices

$$z^i_{s_i} \in Z(i) \cap S, \quad z^j_{s_j} \in Z(j) \cap S, \quad \text{and } r^{(i,j)}_{t_i,t_j} \in R(i,j) \cap S,$$

for some $s_i, s_j, t_i, t_j \in [p]$. Again by Claim 21.3(iii), $z^i_{s_i}$ has d neighbors in $Z_{+B}(i) \cap S$, so if $z^i_{s_i} r^{(i,j)}_{t_i,t_j} \in E(G')$, then $z^i_{s_i}$ has $d+1$ neighbors in S, a contradiction with the fact that S is a $(\sigma^*, \mathbb{N}_{\geq x})$ set. Hence, $z^i_{s_i} r^{(i,j)}_{t_i,t_j} \notin E(G')$ and $z^j_{s_j} r^{(i,j)}_{t_i,t_j} \notin E(G')$. By Observation 21.1, we then have that $s_i = t_i$ and $s_j = t_j$. We can conclude that $v^i_{s_i} v^j_{s_j} \in E(G)$ and since the argument holds for any pair of indices i, j, G has a multicolored clique. ⌟

We would like to remark that by the proof of the previous claim, we have established that any $(\sigma^*, \mathbb{N}_{\geq x})$ set S in G' of size k' in fact contains all vertices from B and one vertex from each $Z(i)$ and from each $R(i,j)$. Since this is precisely the shape of the set constructed in the forward direction of the correctness proof, this shows that any $(\sigma^*, \mathbb{N}_{\geq x})$ set of size k' in G' is a $(\{d\}, \{d+1, \ldots, d+k\})$ set (assuming $k \geq 3$).

Claims 21.2 and 21.4 establish the correctness of the reduction. We observe that $|V(G')| = \mathcal{O}(|V(G)| + d^2 \cdot k^2)$ and clearly, G' can be constructed from G in time polynomial in $|V(G)|$, d and k as well. Furthermore, by (2), $||K|| = \mathcal{O}(d^2 \cdot k^2)$ which implies that $||K|| = \mathcal{O}(k^2)$ since d is a fixed constant and the theorem follows. □

By Proposition 20, the previous theorem has the following consequence.

Corollary 22. *For any fixed $d \in \mathbb{N}$ and $x \leq d+1$, the following holds. Let $\sigma^* \subseteq \mathbb{N}_{\leq d}$ with $d \in \sigma^*$. Then,* Max-$(\sigma^*, \mathbb{N}_{\geq x})$ Domination *is* W[1]*-hard parameterized by linear mim-width plus solution size, and the hardness holds even if a corresponding decomposition tree is given.*

5.2 Minimization Problems

In this section we prove W[1]-hardness of minimization versions of several (σ, ρ) problems parameterized by linear mim-width plus solution size. We obtain our results by modifying a reduction from Multicolored Independent Set to Minimum Dominating Set on H-graphs parameterized by solution size plus $||H||$ due to Fomin et al. [13]. In the Multicolored Independent Set problem we are given a graph G and a partition $V_1, \ldots, V_k$ of its vertex set $V(G)$ and the question is whether there is an independent set $\{v_1, \ldots, v_k\} \subseteq V(G)$ in G such that for each $i \in [k]$, $v_i \in V_i$. The W[1]-hardness of this problem follows immediately from the W[1]-hardness of the Multicolored Clique problem.

The Reduction of Fomin et al. [13]. Let G be an instance of Multicolored Independent Set with partition $V_1, \ldots, V_k$ of $V(G)$. Again we can assume that $k \geq 3$ and that $|V_i| = p$ for all $i \in [k]$. If the latter condition does not hold, let $p \coloneqq \max_{i \in [k]} |V_i|$ and for each $i \in [k]$, add $p - |V_i|$ vertices to V_i that are adjacent to all vertices in each V_j where $j \neq i$. It is clear that the resulting instance has a multicolored independent set if and only if the original instance does.

The graph G' of the Minimum Dominating Set instances is obtained as follows. We take the graph G'' as constructed in the proof of Theorem 21, and for each $i \in [k]$, we add a vertex b_i whose model is $\{u_i\}$, i. e. it is adjacent to all vertices in $Z(i)$ and nothing else. We argue that G has a multicolored independent set if and only if G' has a dominating set of size k.

For the forward direction, if G has a multicolored independent set $I \coloneqq \{v^1_{h_1}, \ldots, v^k_{h_k}\}$, then using Observation 21.1, one can verify that $D \coloneqq \{z^1_{h_1}, \ldots, z^k_{h_k}\}$ is a dominating set in G': Clearly, for each $i \in [k]$, the vertices in $Z(i) \cup \{b_i\}$ are dominated by $z^i_{h_i} \in D$. Suppose there is a vertex $r^{(i,j)}_{s,t} \in R(i,j)$ that is not dominated by D, then in particular it is neither adjacent to $z^i_{h_i}$ nor to $z^j_{h_j}$. By Observation 21.1, this implies that G contains the edge $v^i_{h_i} v^j_{h_j}$, a contradiction with the fact that I is an independent set.

For the backward direction, suppose that G' has a dominating set D of size k. Due to the vertices b_i (for $i \in [k]$), we can conclude that for all $i \in [k]$, $D \cap (Z(i) \cup \{b_i\}) \neq \emptyset$. If D contains b_i for some $i \in [k]$, then we can replace b_i by any vertex in $Z(i)$ such that the resulting set is still a dominating set of D, so we can assume that $D = \{z^1_{h_1}, \ldots, z^k_{h_k}\}$. We claim that $\{v^1_{h_1}, \ldots, v^k_{h_k}\}$ is an independent set in G. Suppose that for $i, j \in [k]$, there is an edge $v^i_{h_i} v^j_{h_j} \in E(G)$. Observation 21.1 implies that $r^{(i,j)}_{h_i,h_j}$ is neither adjacent to $z^i_{h_i}$ nor to $z^j_{h_j}$, so $r^{(i,j)}_{h_i,h_j}$ is not dominated by D, a contradiction.

Remark 23. We would like to remark that the above reduction is to the Min-(σ^*, ρ^*) Domination problem, for all $\sigma^* \subseteq \mathbb{N}$ with $0 \in \sigma^*$ and $\rho^* \subseteq \mathbb{N}^+$ with $\{1, 2\} \subseteq \rho^*$.

Proposition 24 ([13])**.** *For $\sigma^* \subseteq \mathbb{N}$ with $0 \in \sigma^*$ and $\rho^* \subseteq \mathbb{N}^+$ with $\{1, 2\} \subseteq \rho^*$,* Min-$(\sigma^*, \rho^*)$ Domination *is* W[1]*-hard on H-graphs parameterized by the number of edges in H plus solution size, and the hardness holds even when an H-representation of the input graph is given.*

Adaption to Total Domination Problems. Recall that the (σ, ρ) formulation for Dominating Set is $(\mathbb{N}, \mathbb{N}^+)$. We now explain how to modify the above reduction to obtain hardness for total dominating set problems where each vertex in the solution has to have at least one neighbor in the solution as well. These problems include TOTAL DOMINATING SET and DOMINATING INDUCED MATCHING, which can be formulated as $(\mathbb{N}^+, \mathbb{N}^+)$ and $(\{1\}, \mathbb{N}^+)$, respectively. The minimization problem of either of them is known to be NP-complete.

Theorem 25. *For $\sigma^* \subseteq \mathbb{N}^+$ with $1 \in \sigma^*$ and $\rho^* \subseteq \mathbb{N}^+$ with $\{1, 2\} \subseteq \rho^*$, MIN-$(\sigma^*, \rho^*)$ DOMINATION is $\mathsf{W}[1]$-hard on H-graphs parameterized by the number of edges in H plus solution size, and the hardness holds even when an H-representation of the input graph is given.*

Proof. We modify the above reduction from MULTICOLORED INDEPENDENT SET as follows. For each $i \in [k]$, we add another vertex c_i to G' which is only adjacent to b_i. We let $B \coloneqq \bigcup_{i\in[k]}\{b_i\}$ and $C \coloneqq \bigcup_{i\in[k]}\{c_i\}$. Note that these new vertices can be 'hardcoded' into H with the number of edges in H increasing only by k. To argue the correctness of the reduction, we now show that G has a multicolored independent set if and only if G' has a (σ^*, ρ^*) set of size $k' \coloneqq 2k$.

For the forward direction, suppose that G has an independent set $\{v^1_{h_1}, \ldots, v^k_{h_k}\}$. Then, $D' \coloneqq \{z^1_{h_1}, \ldots, z^k_{h_k}\}$ dominates all vertices in $V(G') \setminus C$ by the same argument as above and $D \coloneqq D' \cup B$ dominates all vertices of G'. Furthermore, each $x \in D$ has precisely one neighbor in D: For each such x, either $x = z^i_{h_i}$ or $x = b_i$ for some $i \in [k]$. In the former case, $N(x) \cap D = \{b_i\}$ and in the latter case, $N(x) \cap D = \{z^i_{h_i}\}$. Now let $y \in V(G') \setminus D$. If $y \in Z(i) \cup \{c_i\}$ for $i \in [k]$, then $\emptyset \neq N(y) \cap D \subseteq \{z^i_{h_i}, b_i\}$. If $y \in R(i, j)$ for some $1 \le i < j \le k$, then y is either dominated by one of $z^i_{h_i}$ and $z^j_{h_j}$ or by both and it cannot have any other neighbors in D by construction. Since $1 \in \sigma^*$ and $\{1, 2\} \subseteq \rho^*$, D is a (σ^*, ρ^*) set and clearly, $|D| = 2k$.

For the backward direction, suppose that G' has a (σ^*, ρ^*) set D of size $2k$. Let $i \in [k]$. Since $0 \notin \sigma^*$ and $0 \notin \rho^*$, we have that at least one of c_i and b_i is contained in D (either c_i is dominating or it needs to be dominated). Suppose $c_i \in D$. Since each vertex in D has to have at least one neighbor in D and b_i is the only neighbor of c_i, we can conclude that $b_i \in D$. So, in either case, we have that b_i is contained in D and subsequently we have that $B \subseteq D$. Since $0 \notin \sigma^*$, all vertices of B have a neighbor in D. Suppose for some $i \in [k]$ that neighbor is c_i. Then, we can replace c_i with some $z^i_{h_i} \in Z(i)$, without changing the fact that D is a (σ^*, ρ^*) set. We can assume that for each $i \in [k]$, the neighbor of b_i that is contained in D is a vertex $z^i_{h_i} \in Z(i)$. We have that $D = B \cup \{z^1_{h_1}, \ldots, z^k_{h_k}\}$ and since D is a dominating set (in other words, $0 \notin \rho^*$), we can again argue using Observation 21.1 that $\{v^1_{h_1}, \ldots, v^k_{h_k}\}$ is an independent set in G. $\square$

As a somewhat orthogonal result to Theorem 21, we now show hardness of several problems related to the d-DOMINATING SET problem, where each vertex that is not in the solution set has to be dominated by at least some fixed number of d neighbors in the solution.

Adaption to d-Domination Problems. We use a similar gadget as the one constructed in the proof of Theorem 21 to prove hardness of several (σ, ρ) problems where each vertex has to be dominated by at least d vertices. In particular, we prove the following theorem. Note that the analogous statement of the following theorem for $d = 1$ is proved by the reduction explained in the beginning of this section, see Remark 23.

Theorem 26. *For any fixed $d \in \mathbb{N}_{\geq 2}$, the following holds. Let $\sigma^* \subseteq \mathbb{N}$ with $\{0, 1, d-1\} \subseteq \sigma^*$ and $\rho^* \subseteq \mathbb{N}_{\geq d}$ with $\{d, d+1\} \subseteq \rho^*$. Then, MIN-$(\sigma^*, \rho^*)$ DOMINATION is $\mathsf{W}[1]$-hard on H-graphs parameterized by the number of edges in H plus solution size, and the hardness even holds when an H-representation of the input graph is given.*

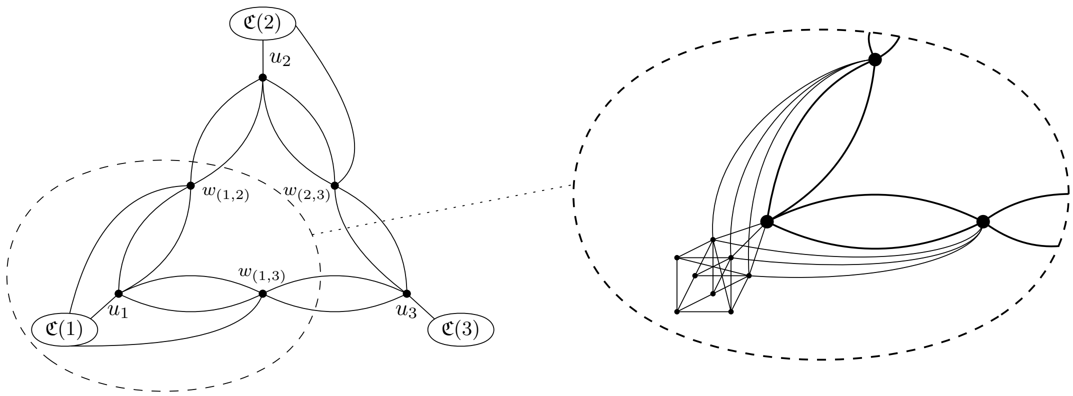


Figure 6: An example graph K w.r.t. which the graph G' constructed in the proof of Theorem 26 is a K-graph. In this example, $k = 3$.

Proof. We modify the reduction from Multicolored Independent Set to Dominating Set on H-graphs due to Fomin et al. [13] that we summarized in the beginning of this section. Let G be a graph with vertex partition $V_1, \ldots, V_k$ and $|V_i| = p$ for all $i \in [k]$ and assume $k \geq 3$. We first describe the gadget we use and then we describe how to construct the graph G' of the Min-(σ^*, ρ^*) Domination instance.

The Gadget $\mathfrak{C}(i)$. Let $i \in [k]$. The gadget $\mathfrak{C}(i)$ is a complete bipartite graph with bipartition $(C_1(i), C_2(i))$ where $C_1(i) := \{c^i_{1,1}, \ldots, c^i_{1,d}\}$ and $C_2(i) := \{c^i_{2,1}, \ldots, c^i_{2,d}\}$ such that each vertex $c^i_{1,j}$ for $j \in [d-1]$ is additionally adjacent to all vertices in $Z(i)$ as well as to all vertices in $R(i,j)$ for $j > i$. (Note that $c^i_{1,d}$ does not have these additional adjacencies.) Throughout the following, we let $C(i) := C_1(i) \cup C_2(i)$ and $C := \bigcup_{i \in [k]} C(i)$.

The graph G' is now obtained by constructing the graph G'' as in the proof of Theorem 21 and then, for each $i \in [k]$, adding the gadget $\mathfrak{C}(i)$ and adding a 'satellite vertex' s_i, adjacent to all vertices in $Z(i) \cup C_1(i)$. G' is a K-graph for the graph $K \supseteq H$, obtained by 'hardcoding' each $\mathfrak{C}(i)$, for $i \in [k]$, into H. That is, for each $i \in [k]$, we add a complete bipartite graph with bipartition $(\{\gamma^i_{1,1}, \ldots, \gamma^i_{1,d}\}, \{\gamma^i_{2,1}, \ldots, \gamma^i_{2,d}\})$, and make all vertices $\gamma^i_{1,h}$, where $h \in [d-1]$, adjacent to u_i as well as to all vertices $w_{(i,j)}$ with $j > i$. For an illustration of K see Figure 6. Note that

$$||K|| = ||H|| + k(d^2 + 1) + \sum_{i=1}^{k} (k - i)(d - 1) = \mathcal{O}(k^2 \cdot d + k \cdot d^2). \tag{4}$$

We illustrate the structure of the graph G' in Figure 7. We now argue that G' is a K-graph. We begin by constructing a subdivision K' of K. First, we do Step K1 that was taken in the proof of Theorem 21 (see page 19) to construct the subdivision, and then the analogue of Step K2 in the proof of Theorem 21 for the gadgets $\mathfrak{C}(i)$. We continue with the following two steps.

(K3) For each $i \in [k]$, let $\{\gamma^i_{1,1}, \ldots, \gamma^i_{1,d}, \gamma^i_{2,1}, \ldots, \gamma^i_{2,d-1}\}$ be the vertices of the copy of the graph of $\mathfrak{C}(i)$ in K. For each $h \in [d-1]$, we subdivide the edge $u_i\gamma^i_{1,h}$ once, and denote the corresponding subdivision node by $s(i,h)$.

(K4) Furthermore, for each $i \in [k]$, for each $i < j \leq k$, and $h \in [d-1]$ we subdivide the edge $w_{(i,j)}\gamma^i_{1,h}$ once, and denote the corresponding subdivision node by $s((i,j),h)$.

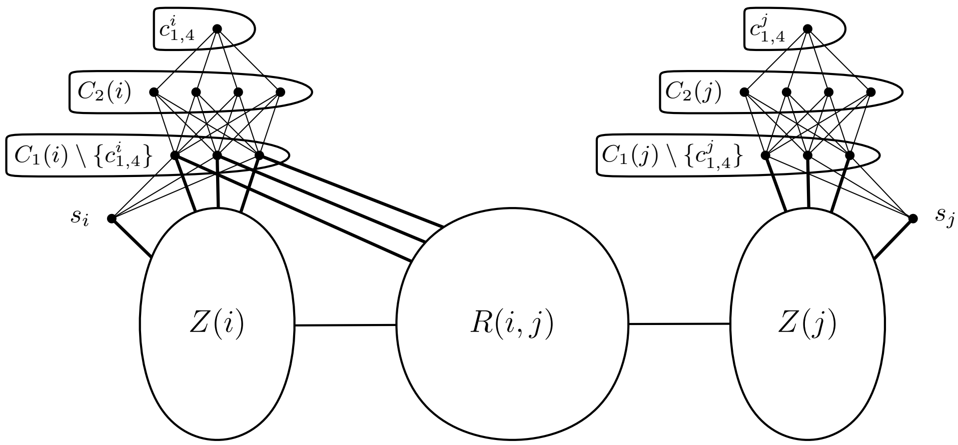


Figure 7: Illustration of a part of G' constructed in the proof of Theorem 26, where $1 \leq i < j \leq k$ and $d = 4$.

We now sketch how to obtain a K-representation of G', $\mathcal{M}' = \{M'_v\}_{v \in V(G')}$, where each M'_v is a connected subset of $V(K')$; for an illustration of G' see Figure 7. As these steps are very similar to Steps (R1) to (R5) in the proof of Theorem 21 (see page 20), we focus on pointing out how to adapt them rather than fully restating all of them.

First, for each $i \in [k]$, the model for the vertex s_i consists of the vertex u_i, i.e. we add the model $M'_{s_i} \coloneqq \{u_i\}$ to $\mathcal{M}'$. Then we do the steps analogous to Steps R1 and R2 taken in the proof of Theorem 21. Following that, we take the steps analogous to R4 and R5, where in Step R4 we consider vertex $c^i_{1,h}$ instead of vertex $b^i_{1,h}$, and in Step R5, we consider vertex $c^i_{1,h}$ instead of vertex $b^{(i,j)}_{1,h}$. Furthermore, in Step R4, we additionally add $s(i,h)$ to the model of s_i. This completes the construction of the K-representation for G'.

Claim 26.1. If G has a multicolored independent set, then G' has a (σ^*, ρ^*) set of size $k' \coloneqq k \cdot (d+1)$.

Proof. Let $\{v^1_{h_1}, \dots, v^k_{h_k}\}$ be the independent set in G. By the reduction proving Proposition 24 in the beginning of this section, we have that $D' \coloneqq \{z^1_{h_1}, \dots, z^k_{h_k}\}$ is a $(\{0\}, \{1,2\})$-set of $G' - C$ of size k (see also Remark 23). Let $C_1 \coloneqq \bigcup_{i \in [k]} C_1(i)$, $C_2 \coloneqq C \setminus C_1$ and $D \coloneqq D' \cup C_1$.

Since each vertex in $V(G') \setminus (D \cup C)$ is adjacent to precisely $d-1$ vertices in C_1 and to either one or two vertices in D' (and $D' \cap C_1 = \emptyset$), we can conclude that each vertex in $V(G') \setminus (D \cup C)$ is adjacent to either d or $d+1$ vertices in D. Since each $C(i)$ induces a $K_{d,d}$, we can conclude that all vertices in C_2 have d neighbors in D as well. Furthermore, $N(s_i) \cap D = (C_1(i) \setminus \{c^i_{1,d}\}) \cup \{z^i_{h_i}\}$, so we have that all vertices in G' that are not contained in D have either d or $d+1$ neighbors in D.

Let $i \in [k]$. Then, $N(z^i_{h_i}) \cap D = \{c^i_{1,1}, \dots, c^i_{1,d-1}\}$, $N(c^i_{1,d}) \cap D = \emptyset$ and for $\ell \in [d-1]$, $N(c^i_{1,\ell}) \cap D = \{z^i_{h_i}\}$. We can conclude that D is a $(\{0,1,d-1\}, \{d,d+1\})$-set in G' and clearly, $|D| = k + kd = k'$. ⌟

In what follows, the strategy is to argue that each (σ^*, ρ^*) set of size $k' = k \cdot (d+1)$ contains a set $\{z^1_{h_1}, \dots, z^k_{h_k}\}$ which will imply that $\{v^1_{h_1}, \dots, v^k_{h_k}\}$ is an independent set in G. Throughout the following, for $i \in [k]$, we let $Z_+(i) \coloneqq Z(i) \cup C(i) \cup \{s_i\}$.

Claim 26.2. For all $i \in [k]$, any (σ^*, ρ^*) set D in G' contains at least d vertices from $C(i)$ and at least $d+1$ vertices from $Z_+(i)$.

Proof. We first show that each such D contains at least d vertices from $C(i)$. Suppose not, then $|D \cap C(i)| \le d-1$ for some $i \in [k]$. If $c^i_{1,d} \notin D$, then $C_2(i) \subseteq D$, otherwise $c^i_{1,d}$ cannot have d or more neighbors in D. But $|C_2(i)| = d$, a contradiction. We can assume that $c^i_{1,d} \in D$. Furthermore, there is at least one vertex $c^i_{2,\ell}$ for $\ell \in [d]$ with $c^i_{2,\ell} \notin D$. To ensure that $c^i_{2,\ell}$ has at least d neighbors in D, we would have to include all remaining vertices from $C_1(i)$ in D, but then $|D \cap C(i)| \ge d$, a contradiction. The second part of the claim now follows since the vertex s_i only has neighbors in $Z_+(i)$ and at most $d-1$ neighbors in $C(i) \cap D$ (namely $C_1(i) \setminus \{c^i_{1,d}\}$): Since D is a (σ^*, ρ^*) set, it either has to contain s_i or at least one additional neighbor of s_i. ⌟

Claim 26.3. For all $i \in [k]$, any (σ^*, ρ^*) set D of size at most $k' = k(d+1)$ contains $C_1(i)$. We furthermore can assume that it additionally contains some $z^i_{h_i} \in Z(i)$, where $h_i \in [p]$.

Proof. By Claim 26.2 we have that D contains $d+1$ vertices from each $Z_+(i')$, $i' \in [k]$, and no other vertices. Consider any vertex $z^i_s \in Z(i)$ (where $s \in [p]$) that is not contained in D. Recall that z^i_s has to have at least d neighbors in D. By Claim 26.2, z^i_s has precisely one neighbor in $(Z(i) \cup \{s_i\}) \cap D$ and since D does not contain any vertex from any $R(j,i)$ $(1 \le j < i)$ or $R(i,j')$ $(i < j' \le k)$, the only possible neighbors of z^i_s in D are $(C_1(i) \cup \{s_i\}) \setminus \{c^i_{1,d}\}$. Furthermore, we observe that $c^i_{1,d} \in D$: for if $c^i_{1,d} \notin D$, then $c^i_{1,d}$ has to have either d or $d+1$ neighbors in D. However, D already contains the $d-1$ vertices $C_1(i) \setminus \{c^i_{1,d}\}$ that are not adjacent to $c^i_{1,d}$, and D contains $d+1$ vertices from $Z_+(i)$. So, at most two neighbors of $c^i_{1,d}$ are contained in D. If at most one neighbor of $c^i_{1,d}$ is contained in D, this immediately gives a contradiction with D being a (σ^*, ρ^*) set since $d \ge 2$. However, if $d = 2$, and two neighbors of $c^i_{1,d}$ are contained in D, then each vertex in $Z(i)$ has only $d-1$ neighbors in D, namely the ones in $C_1(i) \setminus \{c^i_{1,d}\}$, again a contradiction with D being a (σ^*, ρ^*) set. We can conclude that $C_1(i) \subseteq D$.

Now suppose that $s_i \in D$. Then, after swapping s_i with any vertex in $Z(i)$, the resulting set remains a (σ^*, ρ^*) set: Clearly, the condition of being a (σ^*, ρ^*) set is not violated by any vertex in $Z_+(i)$. For $i < j \le k$, consider any vertex $x \in R(i,j)$. Then, $N(x) \cap D$ contains the $d-1$ vertices $C_1(i) \setminus \{c^i_{1,d}\}$, and at most one more each from $Z(i)$ and $Z(j)$, as D can contain at most one vertex from each $Z(i')$, $i' \in [k]$. Now, if we swapped s_i with some vertex from $Z(i)$, then this means that initially, D contained d neighbors from $N(x)$, namely $C_1(i) \setminus \{c^i_{1,d}\}$ and one vertex from $Z(j)$. Hence, after swapping, D contains $d+1$ vertices and since $d+1 \in \rho^*$, D remained a (σ^*, ρ^*) set. An analogous argument can be given for any $R(j', i)$, where $1 \le j' < i$. ⌟

We are now ready to conclude the correctness proof of the reduction.

Claim 26.4. If G' has a (σ^*, ρ^*) set of size $k' = k(d+1)$, then G has a multicolored independent set.

Proof. Let D be a (σ^*, ρ^*) set of size k'. By Claim 26.3, we can assume that $D = C_1 \cup \{z^1_{h_1}, \ldots, z^k_{h_k}\}$ for some $h_1, \ldots, h_k \in [p]$. Now, since for each $1 \le i < j \le k$, all vertices in $R(i,j)$ have precisely $d-1$ neighbors in C_1, each of them has to have at least one of $z^i_{h_i}$ and $z^j_{h_j}$ as a neighbor. By Observation 21.1, this allows us to conclude that $\{v^1_{h_1}, \ldots, v^k_{h_k}\}$ is an independent set in G. ⌟

Claims 26.1 and 26.4 establish the correctness of the reduction. Clearly, $|V(G')| = \mathcal{O}(|V(G)| + d^2 \cdot k)$ (and G' can be constructed in polynomial time) and by (4), $||K|| = \mathcal{O}(k^2 \cdot d + k \cdot d^2)$. Since d is a fixed constant we have that $||K|| = \mathcal{O}(k^2)$ and the theorem follows. □

Similarly to above, a combination of the previous two theorems with Proposition 20 yields the following hardness results for (σ, ρ) mimization problems on graphs of bounded linear mim-width.

Corollary 27. *Let $\sigma^* \subseteq \mathbb{N}$ and $\rho^* \subseteq \mathbb{N}$. Then,* Min-$(\sigma^*, \rho^*)$ Domination *is* $\mathsf{W}[1]$*-hard parameterized by linear mim-width plus solution size, if one of the following holds.*

(i) $\sigma^* \subseteq \mathbb{N}^+$ *with* $1 \in \sigma^*$ *and* $\rho^* \subseteq \mathbb{N}^+$ *with* $\{1, 2\} \subseteq \rho^*$.

(ii) For some fixed $d \in \mathbb{N}_{\geq 2}$, $\{0, 1, d-1\} \subseteq \sigma^*$ *and* $\rho^* \subseteq \mathbb{N}_{\geq d}$ *with* $\{d, d+1\} \subseteq \rho^*$.

Furthermore, the hardness holds even if a corresponding decomposition tree is given.

6 Conclusion

We have introduced the class of distance-r (σ, ρ) and LCVP problems. This generalizes well-known graph distance problems like distance-r domination, distance-r independence, distance-r coloring and perfect r-codes. It also introduces many new distance problems for which the standard distance-1 version naturally captures a well-known graph property.

Using the graph parameter mim-width, we showed that all these problems are solvable in polynomial time for many interesting graph classes. These meta-algorithms will have runtimes which can likely be improved for a particular problem on a particular graph class. For instance, blindly applying our results to solve Distance-r Dominating Set on permutation graphs yields an algorithm that runs in time $\mathcal{O}(n^8)$: Permutation graphs have linear mim-width 1 (with a corresponding decomposition tree that can be computed in linear time) [3, Lemmas 2 and 5], so we can apply Corollary 15(i). However, there is an algorithm that solves Distance-r Dominating Set on permutation graphs in time $\mathcal{O}(n^2)$ [26]; a much faster runtime.

Recently, Chiarelli et al. [10] gave algorithms for the (Total) k-Dominating Set problems on proper interval graphs that run in time $\mathcal{O}(n^{3k})$. The mim-width framework yields an algorithm for these problems that runs in time $\mathcal{O}(n^{2k+4})$ which follows from Proposition 3(i) and the result that interval graphs have linear mim-width 1 [3]. Hence the work [10] improves the generic mim-width based algorithm whenever $k < 4$. We would like to remind the reader however that prior to this work, the result formulated in Proposition 3(i) has not been explicitly stated anywhere.

Regarding lower bounds, we expanded on the previous results by Fomin et al. [13] and showed that many (σ, ρ) problems are $\mathsf{W}[1]$-hard parameterized by mim-width. However, it remains open whether there exists a problem which is NP-hard in general, yet FPT parameterized by mim-width. In particular, several (σ, ρ) problems are not covered by the $\mathsf{W}[1]$-hardness results of Fomin et al. [13] and the ones presented in this paper. Examples include Perfect Code and Perfect Dominating Set, see e.g. Table 1. Even so, we conjecture that every NP-hard (distance) (σ, ρ) problem is $\mathsf{W}[1]$-hard parameterized by mim-width.

Somewhat surprisingly, we proved that powers of graphs of bounded tree-width or clique-width have bounded mim-width. Heggernes et al. [14] showed that the clique-width of the k-th power of a path of length $k(k+1)$ is exactly k. This also shows that the expressive power of mim-width is much stronger than clique-width, since all powers of paths have mim-width just 1. As a special case, we show that leaf power graphs have mim-width 1. We believe the notion of mim-width can be of benefit to the study of leaf power graphs. We remark that it is a big open problem whether leaf power graphs can be recognized in polynomial time [6, 8, 22, 24]. Knowing that leaf powers have mim-width 1, we can connect this open problem to the open problem regarding the recognition of graphs of bounded mim-width which has been repeatedly stated (e.g. [18, 31]): A polynomial-time algorithm that recognizes graphs of mim-width 1 could prove itself useful in devising a recognition algorithm for leaf power graphs.

Acknowledgements. We would like to thank the anonymous reviewers whose numerous comments improved the quality of the presentation in this paper.

References

[1] Geir Agnarsson, Peter Damaschke, and Magnús M. Halldórsson. Powers of geometric intersection graphs and dispersion algorithms. *Discrete Appl. Math.*, 132(1-3):3–16, 2003.

[2] Gábor Bacsó, Dániel Marx, and Zsolt Tuza. H-free graphs, independent sets, and subexponential-time algorithms. In *Proc. IPEC '16*, volume 63 of *LIPIcs*, pages 3:1–3:12. Schloss Dagstuhl, 2017.

[3] Rémy Belmonte and Martin Vatshelle. Graph classes with structured neighborhoods and algorithmic applications. *Theor. Comput. Sci.*, 511:54 – 65, 2013.

[4] Norman Biggs. Perfect codes in graphs. *J. Combin. Theory, Ser. B*, 15(3):289 – 296, 1973.

[5] A. Brandstädt, V. Le, and J. Spinrad. *Graph Classes: A Survey.* SIAM, 1999.

[6] Andreas Brandstädt. On leaf powers. Technical report, University of Rostock, 2010.

[7] 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.

[8] Tiziana Calamoneri and Blerina Sinaimeri. Pairwise compatibility graphs: A survey. *SIAM Review*, 58(3):445–460, 2016.

[9] Steven Chaplick, Martin Töpfer, Jan Voborník, and Peter Zeman. On H-topological intersection graphs. In *Proc. WG '17*, volume 10520 of *LNCS*, pages 167–179. Springer, 2017.

[10] Nina Chiarelli, Tatiana Romina Hartinger, Valeria Alejandra Leoni, Maria Inés Lopez Pujato, and Martin Milanič. Improved algorithms for k-domination and total k-domination in proper interval graphs. In *Proc. ISCO '18*, pages 290 – 302, 2018.

[11] Michael R. Fellows, Danny Hermelin, Frances A. Rosamond, and Stéphane Vialette. On the parameterized complexity of multiple-interval graph problems. *Theor. Comput. Sci.*, 410(1):53–61, 2009.

[12] Carsten Flotow. On powers of m-trapezoid graphs. *Discrete Appl. Math.*, 63(2):187 – 192, 1995.

[13] Fedor V. Fomin, Petr A. Golovach, and Jean-Florent Raymond. On the tractability of optimization problems on H-graphs. In *Proc. ESA '18*, volume 112 of *LIPIcs*, pages 30:1 – 30:14. Schloss Dagstuhl, 2018.

[14] Pinar Heggernes, Daniel Meister, Charis Papadopoulos, and Udi Rotics. Clique-width of path powers. *Discrete Applied Mathematics*, 205:62–72, 2016.

[15] M. A. Henning, Ortrud R. Oellermann, and Henda C. Swart. Bounds on distance domination parameters. *J. Combin. Inform. Syst. Sci.*, 16(1):11–18, 1991.

[16] Lars Jaffke, O-joung Kwon, Torstein J. F. Strømme, and Jan Arne Telle. Generalized distance domination problems and their complexity on graphs of bounded mim-width. In *Proc. IPEC '18*, volume 115 of *LIPIcs*, pages 6:1–6:13. Schloss Dagstuhl, 2018.

[17] Lars Jaffke, O-joung Kwon, and Jan Arne Telle. Polynomial-time algorithms for the longest induced path and induced disjoint paths problems on graphs of bounded mim-width. In *Proc. IPEC '17*, volume 89 of *LIPIcs*, pages 21:1–21:13. Schloss Dagstuhl, 2017.

[18] Lars Jaffke, O-joung Kwon, and Jan Arne Telle. Mim-width I. Induced path problems. *Discrete Applied Mathematics*, 2019. doi:https://doi.org/10.1016/j.dam.2019.06.026.

[19] Lars Jaffke, O-joung Kwon, and Jan Arne Telle. Mim-width II. The feedback vertex set problem. *Algorithmica*, 2019. doi:https://doi.org/10.1007/s00453-019-00607-3.

[20] Lars Jaffke, O-joung Kwon, and Jan Arne Telle. A unified polynomial-time algorithm for feedback vertex set on graphs of bounded mim-width. In *Proc. STACS '18*, volume 96 of *LIPIcs*, pages 42:1–42:14. Schloss Dagstuhl, 2018.

[21] Ton Kloks. *Treewidth: Computations and approximations*, volume 842 of *LNCS*. Springer, 1994.

[22] Manuel Lafond. On strongly chordal graphs that are not leaf powers. In *Proc. WG '17*, volume 10520 of *LNCS*, pages 386–398. Springer, 2017.

[23] Stefan Mengel. Lower bounds on the mim-width of some graph classes. *Discrete Applied Mathematics*, 248:28–32, 2018.

[24] Ragnar Nevries and Christian Rosenke. Towards a characterization of leaf powers by clique arrangements. *Graphs and Combinatorics*, 32(5):2053–2077, 2016.

[25] Krzysztof Pietrzak. On the parameterized complexity of the fixed alphabet shortest common supersequence and longest common subsequence problems. *J. Comput. Syst. Sci.*, 67(4):757–771, 2003.

[26] Akul Rana, Anita Pal, and Madhumangal Pal. An efficient algorithm to solve the distance k-domination problem on permutation graphs. *J. Discrete Math. Sci. Cryptography*, 19(2):241–255, 2016.

[27] Arundhati Raychaudhuri. On powers of strongly chordal and circular arc graphs. *Ars Combin.*, 34:147–160, 1992.

[28] Sigve Hortemo Sæther and Martin Vatshelle. Hardness of computing width parameters based on branch decompositions over the vertex set. *Theor. Comput. Sci.*, 615:120–125, 2016.

[29] Peter J. Slater. R-domination in graphs. *J. ACM*, 23(3):446–450, 1976.

[30] Jan Arne Telle and Andrzej Proskurowski. Algorithms for vertex partitioning problems on partial k-trees. *SIAM J. Discrete Math.*, 10(4):529–550, 1997.

[31] Martin Vatshelle. *New width parameters of graphs*. PhD thesis, University of Bergen, Norway, 2012.