

# Classes of intersection digraphs with good algorithmic properties

## Abstract

While intersection graphs play a central role in the algorithmic analysis of hard problems on undirected graphs, the role of intersection *digraphs* in algorithms is much less understood. We present several contributions towards a better understanding of the algorithmic treatment of intersection digraphs. First, we introduce natural classes of intersection digraphs that generalize several classes studied in the literature. Second, we define the directed locally checkable vertex (DLCV) problems, which capture many well-studied problems on digraphs such as (INDEPENDENT) DOMINATING SET, KERNEL, and  $H$ -HOMOMORPHISM. Third, we give a new width measure of digraphs, *bi-mim-width*, and show that the DLCV problems are polynomial-time solvable when we are provided a decomposition of small bi-mim-width. Fourth, we show that several classes of intersection digraphs have bounded bi-mim-width, implying that we can solve all DLCV problems on these classes in polynomial time given an intersection representation of the input digraph. We identify reflexivity as a useful condition to obtain intersection digraph classes of bounded bi-mim-width, and therefore to obtain positive algorithmic results.

## 1 Introduction

The computational intractability of graph problems is often dealt with by restricting the input graph to be a member of some graph class and exploit the structural properties of this class to design efficient algorithms. Intersection graph classes are an extensively studied family of classes of undirected graphs where vertices are represented by sets with two vertices being adjacent if and only if their corresponding sets intersect. For instance, a graph is an *interval graph* if it is an intersection graph of intervals on a line. The literature on algorithmic aspects of classes of intersection graphs is vast, and we refer to [13] for an overview. Even though the concept of intersection *digraphs* has already been introduced in the early 1980s [9], these classes of directed graphs have not received nearly as much attention in the algorithmic literature as their undirected counterparts. That is not to say that they have not been considered before; for instance, interval digraphs [41], circular-arc digraphs [42], and permutation digraphs [34] have been introduced quite early on.

Formally, a digraph  $G$  is an *intersection digraph* if there exists a family  $\{(S_v, T_v) : v \in V(G)\}$  of ordered pairs of sets such that there is an edge from  $v$  to  $w$  in  $G$  if and only if  $S_v$  intersects  $T_w$ . Note that we add a loop on a vertex  $v$  if  $S_v$  and  $T_v$  intersect. Even for interval digraphs, a natural starting point for the investigation of algorithmic properties of intersection digraphs, no algorithmic applications are known besides a polynomial-time recognition algorithm of the class [34]. One possible explanation for this is that the class of interval digraphs appears to be much richer than their undirected counterparts. We observe that interval digraphs contain, for each integer  $n$ , some orientation of the  $(n \times n)$ -grid (see Proposition 4.5); in contrast, interval graphs do not contain an induced subgraph isomorphic to the 1-subdivision of the claw. This shows that the underlying undirected graphs of interval digraphs are very different from interval graphs.

The case of interval digraphs suggests that further structural restrictions are necessary to make classes of intersection digraphs amenable for algorithmic treatment. In this vein, restrictions of interval digraphs have been considered in the literature [20, 38] with applications to digraph problems such as INDEPENDENT DOMINATING SET, KERNEL, and LIST HOMOMORPHISM. A common feature of the restrictions considered in [20, 38] is that the digraphs are *reflexive*, meaning that each vertex has a loop. Note that for a class of intersection digraphs, reflexivity gives much more additional structure than just added loops.

In this work, we give a host of algorithmic applications of intersection digraph classes, in the following manner:

- We give new and more general classes of intersection digraphs, namely  $H$ -digraphs, rooted directed path digraphs, and  $H$ -convex digraphs. (See the discussion below Theorem 3 for definitions.)
- We introduce directed analogues of the locally checkable vertex problems [46], which include many well-studied digraph problems such as (INDEPENDENT) DOMINATING SET, KERNEL,  $H$ -HOMOMORPHISM, and ORIENTED  $k$ -COLORING, see Tables 1 and 2.
- We define a new width measure of digraphs, called *bi-mim-width*, and prove that the directed locally checkable vertex problems can be solved in polynomial time when a decomposition of bounded bi-mim-width of the input graph is given.
- We prove that fairly general subclasses of these intersection digraph classes have bounded bi-mim-width, see Figure 1.

Note in particular that the last item implies that given a representation of the input digraph, all directed locally checkable problems are solvable in polynomial time on the classes of intersection digraphs in question. For  $H$ -digraphs, we identify reflexivity as the additional restriction that gives bounded bi-mim-width, and therefore algorithmic applications, while we prove that the bi-mim-width is unbounded when we drop this requirement. Recently, Francis, Hell, and Jacob [22] obtained polynomial-time algorithms for KERNEL, DOMINATING SET, and ABSORBING SET on reflexive interval digraphs. Our results are more general in two ways: we give algorithms for more problems, including the aforementioned ones (see Tables 1 and 2), and on much broader digraph classes (see Figure 1). In the following, we discuss the above items in more detail.

**Bi-mim-width.** We introduce a new digraph width parameter, called *bi-mim-width*, which is a directed analogue of the mim-width of an undirected graph introduced by Vatschelle [47]. Roughly speaking, the bi-mim-width of a digraph  $G$  is defined as a branch-width with a cut function that measures for a vertex partition  $(A, B)$  of  $G$ , the sum of the sizes of maximum induced matchings in two bipartite digraphs, one induced by edges from  $A$  to  $B$ , and the other induced by edges from  $B$  to  $A$ . This is similar to how rank-width is generalized to bi-rank-width for digraphs [31, 32]. We formally define bi-mim-width and linear bi-mim-width in Section 3. We compare bi-mim-width and other known width parameters. The mim-width of an undirected graph is exactly the half of the bi-mim-width of the digraph obtained by replacing each edge with bi-directed edges, and this observation can be used to argue that a bound on the bi-mim-width of a class of digraphs implies a bound on the mim-width of a certain class of undirected graphs.

**Directed Locally Checkable Vertex (DLCV) Problems.** We introduce directed locally checkable vertex subset (DLCVS) and partitioning (DLCVP) problems, in analogy with [46]. We

$\sigma^+$	$\sigma^-$	$\rho^+$	$\rho^-$	Standard name
$\{0\}$	$\{0\}$	$\mathbb{N} \setminus \{0\}$	$\mathbb{N}$	Kernel [48]
$\{0, \dots, k-1\}$	$\{0\}$	$\{i : i \geq l\}$	$\mathbb{N}$	$(k, l)$ -out Kernel [39]
$\mathbb{N}$	$\mathbb{N}$	$\mathbb{N}$	$\mathbb{N} \setminus \{0\}$	Dominating set [25]
$\{0\}$	$\{0\}$	$\mathbb{N}$	$\mathbb{N} \setminus \{0\}$	Independent Dominating set [16]
$\mathbb{N}$	$\mathbb{N}$	$\mathbb{N} \setminus \{0\}$	$\mathbb{N}$	In-Dominating set/Absorbing set [23]
$\mathbb{N}$	$\mathbb{N}$	$\mathbb{N} \setminus \{0\}$	$\mathbb{N} \setminus \{0\}$	Twin Dominating set [17]
$\mathbb{N}$	$\mathbb{N}$	$\mathbb{N}$	$\{i : i \geq k\}$	$k$ -Dominating set [35]
$\mathbb{N}$	$\mathbb{N} \setminus \{0\}$	$\mathbb{N}$	$\mathbb{N} \setminus \{0\}$	Total Dominating set [2]
$\{0\}$	$\{0\}$	$\mathbb{N}$	$\{1\}$	Efficient (Closed) Dominating set [8]
$\mathbb{N}$	$\{1\}$	$\mathbb{N}$	$\{1\}$	Efficient Total Dominating set [40]
$\{k\}$	$\{k\}$	$\mathbb{N}$	$\mathbb{N}$	$k$ -Regular Induced Subdigraph [15]

Table 1: Examples of  $(\sigma^+, \sigma^-, \rho^+, \rho^-)$ -sets. For any row there is an associated NP-complete problem, usually maximizing or minimizing the cardinality of a set with the property. Some properties are known under different names; e.g. Efficient Total Dominating sets are also called Efficient Open Dominating sets, and here even the existence of such a set in a digraph  $G$  is NP-complete, as it corresponds to deciding if  $V(G)$  can be partitioned by the open out-neighborhoods of some  $S \subseteq V(G)$ . If rows A and B have their in-restrictions and out-restrictions swapped for both  $\sigma$  and  $\rho$  (i.e.  $\sigma^+$  of row A equals  $\sigma^-$  of row B and vice-versa, and same for  $\rho^+$  and  $\rho^-$ ), then a row-A set in  $G$  is always a row-B set in the digraph with all arcs of  $G$  reversed; this is the case for Dominating set vs in-Dominating set and for Kernel vs Independent Dominating set.

abbreviate the union of these two families of problems to “DLCV problems”. A DLCVS problem is represented as a  $(\sigma^+, \sigma^-, \rho^+, \rho^-)$ -problem for some  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$ , and it asks to find a maximum or minimum vertex set  $S$  in a digraph  $G$  such that for every vertex  $v$  in  $S$ , the numbers of out/in-neighbors in  $S$  are contained in  $\sigma^+$  and  $\sigma^-$ , respectively, and for every vertex  $v$  in  $V(G) \setminus S$ , the numbers of out/in-neighbors in  $S$  are contained in  $\rho^+$  and  $\rho^-$ , respectively. See Table 1 for several examples that appear in the literature. In particular, it includes the KERNEL problem, which was introduced by von Neumann and Morgenstern [48].

A DLCVP problem is represented by a  $(q \times q)$ -matrix  $D$  for some positive integer  $q$ , where for all  $i, j \in \{1, \dots, q\}$ ,  $D[i, j] = (\mu_{i,j}^+, \mu_{i,j}^-)$  for some  $\mu_{i,j}^+, \mu_{i,j}^- \subseteq \mathbb{N}$ . The problem asks to find a vertex partition of a given digraph into  $X_1, X_2, \dots, X_q$  such that for all  $i, j \in [q]$ , the numbers of out/in-neighbors of a vertex of  $X_i$  in  $X_j$  are contained in  $\mu_{i,j}^+$  and  $\mu_{i,j}^-$ , respectively. DIRECTED  $H$ -HOMOMORPHISM is a directed LCVP problem: For a digraph  $H$  on vertices  $\{1, \dots, q\}$ , we can view a homomorphism from a digraph  $G$  to  $H$  as a  $q$ -partition  $(X_1, \dots, X_q)$  of  $V(G)$  such that we can only have an edge from  $X_i$  to  $X_j$  if the edge  $(i, j)$  is present in  $H$ . See Table 2. The ORIENTED  $k$ -COLORING problem, introduced by Sopena [44], asks whether there is a homomorphism to some orientation of a complete graph on at most  $k$  vertices, and can therefore be reduced to a series of directed LCVP problems. Removing the requirement that the color classes have to be independent sets, Smolíková [43] introduced the notion of a *simple  $k$ -coloring*, requiring however that the number of colors is at least two, to avoid trivial solutions. Several works in the literature concern problems of 2-partitioning the vertex sets of digraphs into parts with degree constraints either inside or between the parts of the partition [1, 3, 4, 5, 6]. All of these problems can be observed to be LCVP problems as well, see Table 2. Note that in the DLCVP-framework, we can consider  $q$ -partitions for any fixed  $q \geq 2$ , for all problems apart from 2-OUT-COLORING. This fails for  $q$ -OUT-COLORING, since this problem asks for a  $q$ -coloring with no monochromatic out-neighborhood.

Problem name	$q$	DLCVP ( $q \times q$ )-matrix $D$
Directed $H$ -Homomorphism [26]	$ V(H) $	$\forall (i, j) \in E(H): D[i, j] = (\mathbb{N}, \mathbb{N})$ $\forall (i, j) \notin E(H): D[i, j] = (\{0\}, \{0\})$
Oriented $k$ -Coloring [18, 45]	$k$	$\bigvee_{H: \vec{K}_k} \text{Directed } H\text{-Homomorphism}$
Simple $k$ -Coloring (*) [43]	$k$	$\bigvee_{H: \vec{K}_k^\circ} \text{Directed } H\text{-Homomorphism}$
$\exists (\sigma^+, \sigma^-, \rho^+, \rho^-)$ -set [This paper]	2	$\begin{pmatrix} (\sigma^+, \sigma^-) & (\mathbb{N}, \mathbb{N}) \\ (\rho^+, \rho^-) & (\mathbb{N}, \mathbb{N}) \end{pmatrix}$
$(\delta^+ \geq k_1, \delta^- \geq k_2)$ -Partition [6]	2	$\begin{pmatrix} (\{j: j \geq k_1\}, \mathbb{N}) & (\mathbb{N}, \mathbb{N}) \\ (\mathbb{N}, \mathbb{N}) & (\mathbb{N}, \{j: j \geq k_2\}) \end{pmatrix}$
$(\delta^+ \geq k_1, \delta^+ \geq k_2)$ -Partition [5]	2	$\begin{pmatrix} (\{j: j \geq k_1\}, \mathbb{N}) & (\mathbb{N}, \mathbb{N}) \\ (\mathbb{N}, \mathbb{N}) & (\{j: j \geq k_2\}, \mathbb{N}) \end{pmatrix}$
$(\Delta^+ \leq k_1, \Delta^+ \leq k_2)$ -Partition [3]	2	$\begin{pmatrix} (\{j: j \leq k_1\}, \mathbb{N}) & (\mathbb{N}, \mathbb{N}) \\ (\mathbb{N}, \mathbb{N}) & (\{j: j \leq k_2\}, \mathbb{N}) \end{pmatrix}$
$(\delta^+ \geq k_1, \delta^- \geq k_2)$ -Bipartite-Partition [4]	2	$\begin{pmatrix} (\mathbb{N}, \mathbb{N}) & (\{j: j \geq k_1\}, \mathbb{N}) \\ (\mathbb{N}, \{j: j \geq k_2\}) & (\mathbb{N}, \mathbb{N}) \end{pmatrix}$
$(\delta^+ \geq k_1, \delta^+ \geq k_2)$ -Bipartite-Partition [4]	2	$\begin{pmatrix} (\mathbb{N}, \mathbb{N}) & (\{j: j \geq k_1\}, \mathbb{N}) \\ (\{j: j \geq k_2\}, \mathbb{N}) & (\mathbb{N}, \mathbb{N}) \end{pmatrix}$
2-Out-Coloring [1]	2	$\begin{pmatrix} (\mathbb{N} \setminus \{0\}, \mathbb{N}) & (\mathbb{N} \setminus \{0\}, \mathbb{N}) \\ (\mathbb{N} \setminus \{0\}, \mathbb{N}) & (\mathbb{N} \setminus \{0\}, \mathbb{N}) \end{pmatrix}$

Table 2: Examples of directed LCVP problems. For every row there are choices of values for which the problems are NP-complete. For Directed  $H$ -Homomorphism let  $V(H) = \{1, \dots, |V(H)|\}$  and denote by  $H: \vec{K}_k$  that  $H$  is an orientation of a complete graph on  $k$  vertices, and by  $H: \vec{K}_k^\circ$  that  $H$  is an orientation of a complete graph on  $k$  vertices, with loops. (\*) For SIMPLE  $k$ -COLORING, we require two nonempty color classes to avoid trivial solutions. The general algorithm can easily be modified to take this into account.

**Theorem 1.** *Directed LCVS and LCVP problems can be solved in time  $XP$  parameterized by bi-mim-width, when a branch decomposition is given.*

Furthermore, we show that the distance variants of DLCVS problems, for instance DISTANCE- $r$  DOMINATING SET can be solved in polynomial time on digraphs of bounded bi-mim-width. Another natural variant is the  $k$ -KERNEL problem (see [7, Section 8.6.2]), which asks for a kernel in the  $(k - 1)$ -th power of a given digraph. To show this, we prove that the  $r$ -th power of a digraph of bi-mim-width  $w$  has bi-mim-width at most  $rw$  (Lemma 3.9). For undirected graphs, there is a bound that does not depend on  $r$  [28], but we were not able to obtain such a bound for the directed case.

**Theorem 2.** *Distance variants of directed LCVS problems can be solved in time  $XP$  parameterized by bi-mim-width, when a branch decomposition is given.*

**Classes of intersection digraphs and their bi-mim-width.** We provide various classes of digraphs of bounded bi-mim-width. We first summarize our results in the following theorem and give the background below. We illustrate the bounds in Figure 1.

**Theorem 3.** (i) *Given a reflexive interval digraph, one can output a linear branch decomposition of bi-mim-width at most 2 in polynomial time. On the other hand, interval digraphs have unbounded bi-mim-width.*

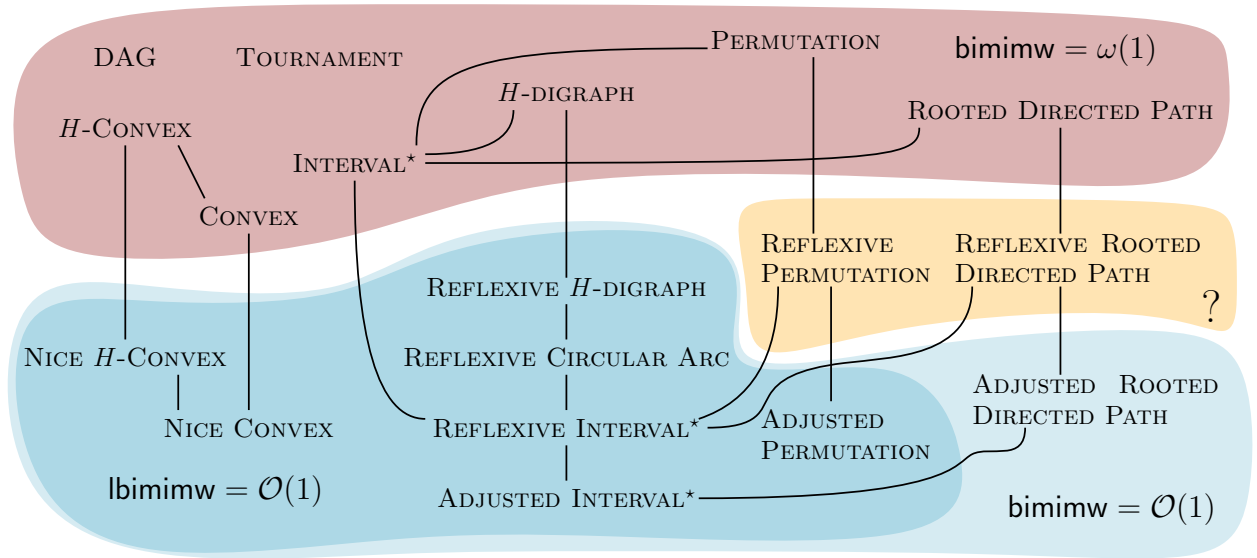


Figure 1: Digraph classes with bounds on their (linear) bi-mim-width. For graph classes marked with \* there are polynomial-time algorithms to compute representations of their members. If digraph class  $A$  is depicted above  $B$  and there is an edge between  $A$  and  $B$  then  $B \subseteq A$ .

- (ii) Given a representation of an adjusted permutation digraph  $G$ , one can construct in polynomial time a linear branch decomposition of  $G$  of bi-mim-width at most 4. Permutation digraphs have unbounded bi-mim-width.
- (iii) Given a representation of an adjusted rooted directed path digraph  $G$ , one can construct in polynomial time a branch decomposition of  $G$  of bi-mim-width at most 2. Rooted directed path digraphs have unbounded bi-mim-width and adjusted rooted directed path digraphs have unbounded linear bi-mim-width.
- (iv) Let  $H$  be an undirected graph. Given a representation of a reflexive  $H$ -digraph  $G$ , one can construct in polynomial time a linear branch decomposition of  $G$  of bi-mim-width at most  $12|E(H)|$ .  $P_2$ -digraphs, which are interval digraphs, have unbounded bi-mim-width.
- (v) Let  $H$  be an undirected graph. Given a nice  $H$ -convex digraph  $G$  with its bipartition  $(A, B)$ , one can construct in polynomial time a linear branch decomposition of  $G$  of bi-mim-width at most  $12|E(H)|$ .  $P_2$ -convex digraphs have unbounded bi-mim-width.
- (vi) Tournaments and directed acyclic graphs have unbounded bi-mim-width.

((i). Interval digraphs) Recall that Müller [34] devised a recognition algorithm for interval digraphs, which also outputs a representation. By testing the reflexivity of a digraph, we can recognize reflexive interval digraphs, and output its representation. We convert it into a linear branch decomposition of bi-mim-width at most 2. On the other hand, interval digraphs generally have unbounded bi-mim-width. By Theorem 1, we can solve all DLCV problems on reflexive interval digraphs in polynomial time. This extends the polynomial-time algorithms for INDEPENDENT DOMINATING SET and KERNEL on interval nest digraphs given by Prisner [38], and includes polynomial-time algorithms for ABSORBING SET, DOMINATING SET, and KERNEL by Francis, Hell, and Jacob [22].

((ii). Permutation digraphs) A *permutation digraph* is an intersection digraph of pairs of line segments whose endpoints lie on two parallel lines. Müller [34] considered permutation digraphs under the name ‘matching diagram digraph’, and observed that every interval digraph is a permutation digraph. Therefore, permutation digraphs have unbounded bi-mim-width. We say that a permutation digraph is *adjusted* if there exists one of the parallel lines, say  $\Lambda$ , such that for all  $v \in V(G)$ ,  $S_v$  and  $T_v$  have the same endpoint in  $\Lambda$ . We show that every adjusted permutation digraph has linear mim-width at most 4.

((iii). Rooted directed path digraphs) It is known that chordal graphs have unbounded mim-width [30, 33]. As restrictions of chordal graphs, it has been shown that rooted directed path graphs, and more generally, leaf power graphs have mim-width at most 1 [28], while they have unbounded linear mim-width. A *rooted directed path digraph* is an intersection digraph of pairs of directed paths in a rooted directed tree (every node is reachable from the root), and it is *adjusted* if for every vertex  $v$ , the endpoint of  $S_v$  that is farther from the root is the same as the endpoint of  $T_v$  that is farther from the root. We show that every adjusted rooted directed path digraph has bi-mim-width at most 2. Since this class includes the biorientations of trees, it has unbounded linear bi-mim-width.

((iv).  $H$ -digraphs) For an undirected graph  $H$ , an  $H$ -graph is an undirected intersection graph of connected subgraphs in an  $H$ -subdivision, introduced by B  r  , Hujter, and Tuza [11]. For example, interval graphs and circular-arc graphs are  $P_2$ -graphs and  $C_3$ -graphs, respectively. Fomin, Golovach, and Raymond [21] showed that  $H$ -graphs have linear mim-width at most  $2|E(H)| + 1$ . Motivated by  $H$ -graphs, we introduce an  $H$ -digraph that is the intersection digraph of pairs of connected subgraphs in an  $H$ -subdivision (where  $H$  and its subdivision are undirected). We prove that reflexive  $H$ -digraphs have linear bi-mim-width at most  $12|E(H)|$ . This extends the linear bound of Fomin et al. [21] for  $H$ -graphs.

((v).  $H$ -convex digraphs) For an undirected graph  $H$ , a bipartite digraph  $G$  with bipartition  $(A, B)$  is an  $H$ -convex digraph, if there exists a subdivision  $F$  of  $H$  with  $V(F) = A$  such that for every vertex  $b$  of  $B$ , each of the set of out-neighbors and the set of in-neighbors of  $b$  induces a connected subgraph in  $F$ . We say that an  $H$ -convex digraph is *nice* if for every vertex  $b$  of  $B$ , there is a bi-directed edge between  $b$  and some vertex of  $A$ . Note that  $H$ -convex graphs, introduced by Bonomo-Braberman et al. [12], can be seen as nice  $H$ -convex digraphs, by replacing every edge with bi-directed edges. We prove that nice  $H$ -convex digraphs have linear bi-mim-width at most  $12|E(H)|$ . This implies that  $H$ -convex graphs have linear mim-width at most  $6|E(H)|$ . For the special case when  $T$  is a tree with maximum degree  $\Delta$  and  $t$  branching nodes, Bonomo-Braberman et al. [12] showed an improved bound of  $2 + t(\Delta - 2)$  on the mim-width of  $T$ -convex graphs.

((vi). Directed acyclic graphs and tournaments) We show that if  $H$  is the underlying undirected graph of a digraph  $G$ , then the bi-mim-width of  $G$  is at least the mim-width of  $H$ . Using this, we can show that acyclic orientations of grids have unbounded bi-mim-width. We also prove that tournaments have unbounded bi-mim-width. This refines an argument that they have unbounded bi-rank-width [7, Lemma 9.9.11].

We can summarize our algorithmic results as follows.

**Corollary 4.** *Given a reflexive interval digraph, or a representation of either an adjusted permutation digraph, or an adjusted rooted directed path digraph, or a reflexive  $H$ -digraph, or a nice  $H$ -convex digraph, we can solve all DLCV problems, and their distance variants, in polynomial time.*



The paper is organized as follows. In Section 2, we introduce basic notations. In Section 3, we formally introduce bi-mim-width and compare with other known width parameters. In Section 4, we prove Theorem 3, and in Section 5, we prove Theorems 1 and 2.

## 2 Preliminaries

For a positive integer  $n$ , we use the shorthand  $[n] := \{1, \dots, n\}$ .

**Undirected Graphs.** We use standard notions of graph theory and refer to [19] for an overview. All undirected graphs considered in this work are finite and simple. For an undirected graph  $G$ , we denote by  $V(G)$  the vertex set of  $G$  and  $E(G)$  the edge set of  $G$ . For an edge  $\{u, v\} \in E(G)$ , we may use the shorthand ‘ $uv$ ’.

For an undirected graph  $G$  and two disjoint vertex sets  $A, B \subseteq V(G)$ , we denote by  $G[A, B]$  the bipartite graph on bipartition  $(A, B)$  such that  $E(G[A, B])$  is exactly the set of edges of  $G$  incident with both  $A$  and  $B$ .

**Digraphs.** All digraphs considered in this work are finite and have no multiple edges, but may have loops. For a digraph  $G$ , we denote by  $V(G)$  its vertex set and by  $E(G) \subseteq V(G) \times V(G)$  its edge set. We say that an edge  $(u, v) \in E(G)$  is directed from  $u$  to  $v$ .

For a digraph  $G$ , the undirected graph obtained by replacing every edge with undirected edge and then removing multiple edges is called its *underlying undirected graph*. For an undirected graph  $G$ , a digraph obtained by replacing every edge  $uv$  with one of  $(u, v)$  and  $(v, u)$  is called its *orientation*, and the digraph obtained by replacing every edge  $uv$  with two directed edges  $(u, v)$  and  $(v, u)$  is called its *biorientation*.

For a digraph  $G$  and two disjoint vertex sets  $A, B \subseteq V(G)$ , we denote by  $G[A \rightarrow B]$  the bipartite digraph on bipartition  $(A, B)$  with edge set  $E(G[A, B]) = E(G) \cap (A \times B)$ , and denote by  $G[A, B]$  the bipartite digraph on bipartition  $(A, B)$  with edge set  $E(G[A \rightarrow B]) \cup E(G[B \rightarrow A])$ . We denote by  $M_G[A \rightarrow B]$  the matrix whose columns are indexed by  $A$  and rows are indexed by  $B$  such that for  $a \in A$  and  $b \in B$ ,  $M_G[A \rightarrow B](a, b) = 1$  if there is an edge from  $a$  to  $b$  and 0 otherwise.

A *tournament* is an orientation of a complete graph.

**Common notations.** Let  $G$  be an undirected graph or a digraph. A set  $M$  of edges in  $G$  is a *matching* if no two edges share an endpoint, and it is an *induced matching* if there are no edges in  $G$  meeting two distinct edges in  $M$ . We denote by  $\nu(G)$  the maximum size of an induced matching of  $G$ .

For two undirected graphs or two directed graphs  $G$  and  $H$ , we denote by  $G \cap H = (V(G) \cap V(H), E(G) \cap E(H))$  and  $G \cup H = (V(G) \cup V(H), E(G) \cup E(H))$ .

For a vertex set  $A$  of  $G$ , we denote by  $\bar{A} := V(G) \setminus A$ . A vertex bipartition  $(A, \bar{A})$  of  $G$  for some vertex set  $A$  of  $G$  will be called a *cut*. A cut  $(A, \bar{A})$  of  $G$  is *balanced* if  $|V(G)|/3 < |A| \leq 2|V(G)|/3$ .

For two vertices  $u, v \in V(G)$ , the *distance* between  $u$  and  $v$ , denoted by  $\text{dist}_G(u, v)$  or simply  $\text{dist}(u, v)$ , is the length of the shortest path from  $u$  to  $v$  (if  $G$  is a digraph, then we consider directed paths). For a positive integer  $d$ , we denote by  $G^d$  the graph obtained from  $G$  by, for every pair  $(x, y)$  of vertices in  $G$ , adding an edge from  $x$  to  $y$  if there is a path of length at most  $d$  from  $x$  to  $y$  in  $G$ . We call it the *d-th power* of  $G$ .

### 3 Bi-mim-width

In this section, we introduce the bi-mim-width of a digraph. For an undirected graph  $G$  and  $A \subseteq V(G)$ , let  $\text{mim}_G(A) := \nu(G[A, \bar{A}])$ . For a digraph  $G$  and  $A \subseteq V(G)$ ,

- let  $\text{mim}_G^+(A) := \nu(G[A \rightarrow \bar{A}])$  and  $\text{mim}_G^-(A) := \nu(G[\bar{A} \rightarrow A])$  and
- let  $\text{bimim}_G(A) := \text{mim}_G^+(A) + \text{mim}_G^-(A)$ .

A tree is *subcubic* if it has at least two vertices and every internal vertex has degree 3. A tree  $T$  is a *caterpillar* if it contains a path  $P$  such that every vertex in  $V(T) \setminus V(P)$  has a neighbor in  $P$ . Let  $G$  be an undirected graph or a digraph. A *branch decomposition* of  $G$  is a pair  $(T, \mathcal{L})$  of a subcubic tree  $T$  and a bijection  $\mathcal{L}$  from  $V(G)$  to the leaves of  $T$ . If  $T$  is a caterpillar, then  $(T, \mathcal{L})$  is called a *linear branch decomposition* of  $G$ .

**Definition 3.1 (Bi-mim-width).** Let  $G$  be a digraph and let  $(T, \mathcal{L})$  be a branch decomposition of  $G$ . For each edge  $e \in E(T)$ , let  $T_A$  and  $T_B$  be the connected components of  $T - e$ . Let  $(A_e, B_e)$  be the cut of  $G$  where  $A_e$  is the set of vertices that  $\mathcal{L}$  maps to the leaves in  $T_A$  and  $B_e$  is the set of vertices that  $\mathcal{L}$  maps to the leaves in  $T_B$ . The *bi-mim-width* of  $(T, \mathcal{L})$  is  $\text{bimimw}(T, \mathcal{L}) := \max_{e \in E(T)} (\text{bimim}_G(A_e))$ . The *bi-mim-width* of  $G$ , denoted by  $\text{bimimw}(G)$ , is the minimum bi-mim-width of any branch decomposition of  $G$ . The *linear bi-mim-width* of  $G$ , denoted by  $\text{lbimimw}(G)$ , is the minimum bi-mim-width of any linear branch decomposition of  $G$ .

This is motivated by the mim-width of an undirected graph introduced by Vatschelle [47].

**Definition 3.2 (Mim-width).** Let  $G$  be an undirected graph and let  $(T, \mathcal{L})$  be a branch decomposition of  $G$ . For each edge  $e \in E(T)$ , let  $T_A$  and  $T_B$  be the two connected components of  $T - e$ . Let  $(A_e, B_e)$  be the cut of  $G$  where  $A_e$  is the set of vertices that  $\mathcal{L}$  maps to the leaves in  $T_A$  and  $B_e$  is the set of vertices that  $\mathcal{L}$  maps to the leaves in  $T_B$ . The *mim-width* of  $(T, \mathcal{L})$  is  $\text{mimw}(T, \mathcal{L}) := \max_{e \in E(T)} \text{mim}_G(A_e)$ . The *mim-width* of  $G$ , denoted by  $\text{mimw}(G)$ , is the minimum mim-width of any branch decomposition of  $G$ . The *linear mim-width* of  $G$ , denoted by  $\text{lmimw}(G)$ , is the minimum mim-width of any linear branch decomposition of  $G$ .

The following two lemmas are clear by definition.

**Lemma 3.3.** *Let  $G$  be a digraph and let  $H$  be an induced subdigraph of  $G$ . Then  $\text{bimimw}(H) \leq \text{bimimw}(G)$  and  $\text{lbimimw}(H) \leq \text{lbimimw}(G)$ .*

**Lemma 3.4.** *Let  $G$  be an undirected graph and let  $H$  be the biorientation of  $G$ . Then for every vertex partition  $(A, B)$  of  $G$ , we have  $\nu(G[A, B]) = \frac{\nu(H[A \rightarrow B]) + \nu(H[B \rightarrow A])}{2}$ . In particular, we have  $\text{mimw}(G) = \frac{\text{bimimw}(H)}{2}$ .*

We show that if a digraph  $G$  has small bi-mim-width, then its underlying undirected graph has small mim-width. But the other direction does not hold; the class of tournaments has unbounded bi-mim-width.

**Lemma 3.5.** *Let  $G$  be a digraph and let  $H$  be the underlying undirected graph of  $G$ . Then  $\text{mimw}(H) \leq \text{bimimw}(G)$  and  $\text{lmimw}(H) \leq \text{lbimimw}(G)$ . On the other hand, the class of tournaments has unbounded bi-mim-width, while their underlying undirected graphs have linear mim-width 1.*



*Proof.* Let  $(T, \mathcal{L})$  be a branch-decomposition of  $H$ , and assume that it has mim-width  $t$ . Then,  $T$  has an edge  $e$  inducing a cut  $(A_e, B_e)$  such that  $\text{mim}_H(A_e) = t$ . Let  $M$  be a maximum induced matching of  $H[A_e, B_e]$ , and let  $(M_1, M_2)$  be the partition of  $M$  such that original edges in  $M_1$  are contained in  $G[A_e \rightarrow B_e]$  and original edges of  $M_2$  are contained in  $G[B_e \rightarrow A_e]$ . It shows that the sum of the sizes of the maximum induced matchings in  $G[A_e \rightarrow B_e]$  and in  $G[B_e \rightarrow A_e]$  is at least  $t$ . This implies that  $(T, \mathcal{L})$  has bi-mim-width at least  $t$ , as a branch-decomposition of  $G$ . As we chose  $(T, \mathcal{L})$  arbitrarily, we conclude that  $G$  has bi-mim-width at least  $\text{mimw}(H)$ . The same argument obtained by replacing a branch-decomposition with a linear branch-decomposition shows that  $\text{lmimw}(H) \leq \text{lbimimw}(G)$ .

We prove the second statement. For every integer  $n \geq 2$ , we define  $G_n$  as the graph on the vertex set  $\{v_{i,j} : i, j \in [n]\}$  satisfying that

- for all  $i \in [n]$  and  $j_1, j_2 \in [n]$  with  $j_1 < j_2$ , there is an edge from  $v_{i,j_1}$  to  $v_{i,j_2}$ ,
- for all  $i \in [n-1]$  and  $j \in [n]$ , there is an edge from  $v_{i,j}$  to  $v_{i+1,j}$ ,
- for all  $i_1, i_2, j_1, j_2 \in [n]$  with  $i_1 < i_2$ ,
  - if  $i_2 - i_1 \equiv 0 \pmod{3}$  and  $j_1 \geq j_2$ , then there is an edge from  $v_{i_1,j_1}$  to  $v_{i_2,j_2}$ ,
  - if  $i_2 - i_1 \equiv 0 \pmod{3}$  and  $j_1 < j_2$ , then there is an edge from  $v_{i_2,j_2}$  to  $v_{i_1,j_1}$ ,
  - if  $i_2 - i_1 \equiv 1 \pmod{3}$  and  $(i_2, j_2) \neq (i_1 + 1, j_1)$ , then there is an edge from  $v_{i_2,j_2}$  to  $v_{i_1,j_1}$ ,
  - if  $i_2 - i_1 \equiv 2 \pmod{3}$ , then there is an edge from  $v_{i_1,j_1}$  to  $v_{i_2,j_2}$ .

For each  $i \in [n]$ , we let  $R_i := \{v_{x,i} : x \in [n]\}$  and  $C_i := \{v_{i,y} : y \in [n]\}$ .

We claim that for every positive integer  $k$ ,  $G_{18k}$  has bi-mim-width at least  $k$ . Suppose for contradiction that there is a branch decomposition of  $G$  of bi-mim-width at most  $k-1$ . Therefore, there is a balanced cut  $(A, B)$  of  $G$  where  $\text{bimim}_G(A) \leq k-1$ . Let  $n := 18k$ .

We divide into two cases.

(Case 1. For every  $i \in [n]$ ,  $R_i$  contains a vertex of  $A$  and a vertex of  $B$ .) Then there exists  $a_i \in [n-1]$  for each  $i \in [n]$  such that  $v_{a_i,i}$  and  $v_{a_i+1,i}$  are contained in distinct sets of  $A$  and  $B$ . Let  $I_1 \subseteq [n]$  be a set of size at least  $n/2$  such that either

- for all  $i \in I_1$ ,  $v_{a_i,i} \in A$ , or
- for all  $i \in I_1$ ,  $v_{a_i,i} \in B$ .

Without loss of generality, we assume that for all  $i \in I_1$ ,  $v_{a_i,i} \in A$ . The proof will be symmetric when  $v_{a_i,i} \in B$ . Furthermore, we take a subset  $I_2$  of  $I_1$  of size at least  $n/6$  such that integers in  $\{a_i : i \in I_2\}$  are pairwise congruent modulo 3.

Now, we verify that  $\{(v_{a_i,i}, v_{a_i+1,i}) : i \in I_2\}$  is an induced matching in  $G[A \rightarrow B]$ . Let  $x, y \in I_2$  be distinct integers and assume that  $a_x \leq a_y$ . If  $a_x = a_y$ , then  $(a_x + 1) - a_y \equiv 1 \pmod{3}$ , and thus there is an edge from  $v_{a_x+1,x}$  to  $v_{a_y,y}$  and similarly, there is an edge from  $v_{a_y+1,y}$  to  $v_{a_x,x}$ . Assume that  $a_x < a_y$ . Then  $a_y - (a_x + 1) \equiv 2 \pmod{3}$  and  $(a_y + 1) - a_x \equiv 1 \pmod{3}$ , and thus there is an edge from  $v_{a_x+1,x}$  to  $v_{a_y,y}$  and there is an edge from  $v_{a_y+1,y}$  to  $v_{a_x,x}$ . This shows that there are no edges between  $\{v_{a_x,x}, v_{a_x+1,x}\}$  and  $\{v_{a_y,y}, v_{a_y+1,y}\}$  in  $G[A \rightarrow B]$ , and therefore  $\{(v_{a_i,i}, v_{a_i+1,i}) : i \in I_2\}$  is an induced matching in  $G[A \rightarrow B]$  of size at least  $n/6 = 3k$ . This contradicts the assumption that  $\text{bimim}_G(A) \leq k-1$ .

(Case 2. For some  $j \in [n]$ ,  $R_j$  is fully contained in one of  $A$  and  $B$ .) Without loss of generality, we assume that  $R_j$  is contained in  $A$ . Since  $|B| > |V(G)|/3$ , there is a subset  $I_1 \subseteq [n]$  such that

$|I_1| \geq n/3$ , and for each  $i \in I_1$ ,  $a_{i,b_i} \in B$  for some  $b_i \in [n]$ . We take a subset  $I_2$  of  $I_1$  of size at least  $n/9$  where all integers in  $I_2$  are pairwise congruent modulo 3. Lastly, we take a subset  $I_3$  of  $I_2$  of size at least  $n/18$  such that either

- for all  $i \in I_3$ ,  $b_i > j$ , or
- for all  $i \in I_3$ ,  $b_i < j$ .

First, we assume that  $b_i > j$  for all  $i \in I_3$ . We verify that  $\{(v_{i,j}, v_{i,b_i}) : i \in I_3\}$  is an induced matching in  $G[A \rightarrow B]$ . Let  $x, y \in I_3$  with  $x < y$ . As  $x$  and  $y$  are congruent modulo 3, there is an edge from  $v_{x,b_x}$  to  $v_{y,j}$  and there is an edge from  $v_{y,b_y}$  to  $v_{x,j}$ . It shows that there are no edges between  $\{v_{x,j}, v_{x,b_x}\}$  and  $\{v_{y,j}, v_{y,b_y}\}$  in  $G[A \rightarrow B]$ , and therefore  $\{(v_{x,j}, v_{x,b_x}) : x \in I_3\}$  is an induced matching in  $G[A \rightarrow B]$  of size at least  $n/18 = k$ . It contradicts the assumption that  $\text{bimim}(A) \leq k - 1$ . The argument when  $b_i < j$  for all  $i \in I_3$  is similar.  $\square$

We argue that directed tree-width [29] and bi-mim-width are incomparable.

**Lemma 3.6.** *Directed tree-width and bi-mim-width are incomparable.*

*Proof.* First, the class of all acyclic orientations of undirected grids has directed tree-width 1 but has unbounded bi-mim-width, by Lemma 3.5. Second, the class of all possible digraphs obtained from a directed cycle by replacing a vertex with a set of vertices with same in-neighbors and out-neighbors that are not adjacent to each other, has bi-mim-width at most 4, but has unbounded directed tree-width. To see that such a digraph has large directed tree-width, one can create a subdivision of a cylindrical grid as a subgraph which has large directed tree-width [29].  $\square$

We compare the bi-mim-width with the bi-rank-width of a digraph, introduced by Kanté [31]. Kanté and Rao [32] later generalized this notion to edge-colored graphs. For a digraph  $G$  and  $A \subseteq V(G)$ ,

- let  $\text{cutrk}_G^+(A) := \text{rank}(M_G[A \rightarrow \bar{A}])$  and  $\text{cutrk}_G^-(A) := \text{rank}(M_G[\bar{A} \rightarrow A])$  and
- let  $\text{bicutrk}_G(A) := \text{cutrk}_G^+(A) + \text{cutrk}_G^-(A)$ ,

where the rank of a matrix is computed over the binary field.

**Definition 3.7 (Bi-rank-width).** Let  $G$  be a digraph and let  $(T, \mathcal{L})$  be a branch decomposition of  $G$ . For each edge  $e \in E(T)$ , let  $T_A$  and  $T_B$  be the two connected components of  $T - e$ . Let  $(A_e, B_e)$  be the cut of  $G$  where  $A_e$  is the set of vertices that  $\mathcal{L}$  maps to the leaves in  $T_A$  and  $B_e$  is the set of vertices that  $\mathcal{L}$  maps to the leaves in  $T_B$ . The *bi-rank-width* of  $(T, \mathcal{L})$  is  $\text{birw}(T, \mathcal{L}) := \max_{e \in E(T)} \text{bicutrk}_G(A_e)$ . The *bi-rank-width* of  $G$ , denoted by  $\text{birw}(G)$ , is the minimum bi-rank-width of any branch decomposition of  $G$ . The *linear bi-rank-width* of  $G$ , denoted by  $\text{lbirw}(G)$ , is the minimum bi-rank-width of any linear branch decomposition of  $G$ .

We can verify that for every digraph  $G$ ,  $\text{bimimw}(G) \leq \text{birw}(G)$ . Interestingly, we can further show that for every positive integer  $r$ , the bi-mim-width of the  $r$ -th power of  $G$  is at most the bi-rank-width of  $G$ . This does not depend on the value of  $r$ .

**Lemma 3.8.** *Let  $r$  and  $w$  be positive integers. If  $(T, \mathcal{L})$  is a branch-decomposition of a digraph  $G$  of bi-rank-width  $w$ , then it is a branch-decomposition of  $G^r$  of bi-mim-width at most  $w$ .*

*Proof.* It is sufficient to prove that for every ordered vertex partition  $(A, B)$  of  $G$ , we have  $\nu(G^r[A \rightarrow B]) \leq \text{rank}(M_G[A \rightarrow B])$ . Assume  $\text{rank}(M_G[A \rightarrow B]) = t$  and suppose for contradiction that  $\nu(G^r[A \rightarrow B]) \geq t + 1$ .

Let  $\{(a_i, b_i) : i \in [t + 1]\}$  be an induced matching of  $G^r[A \rightarrow B]$  with  $\{a_i : i \in [t + 1]\} \subseteq A$ . It means that for each  $i \in [t + 1]$ , there is a directed path  $P_i$  of length at most  $r$  from  $a_i$  to  $b_i$  in  $G$  such that the paths in  $\{P_i : i \in [t + 1]\}$  are pairwise vertex-disjoint. We choose an edge  $(c_i, d_i)$  in each  $P_i$  where  $c_i \in A$  and  $d_i \in B$ . As  $\text{rank}(M_G[\{c_i : i \in [t + 1]\} \rightarrow \{d_i : i \in [t + 1]\}]) \leq \text{rank}(M_G[A \rightarrow B]) = t$ , the matrix  $M_G[\{c_i : i \in [t + 1]\} \rightarrow \{d_i : i \in [t + 1]\}]$  is linearly dependent. In particular, there is a non-empty subset  $I$  of  $[t + 1]$  where the sum of  $M_G[\{c_j\} \rightarrow \{d_i : i \in I\}]$  over  $j \in I$  becomes a zero vector (as we are working on the binary field). We choose such a subset with minimum  $|I|$ . Without loss of generality, we may assume that  $I = [q]$  for some  $q \geq 2$ .

Now, we claim that for each  $x \in [q]$ ,  $c_x$  has an out-neighbor in  $\{d_i : i \in [q] \setminus \{x\}\}$ . Suppose that this is not true, that is, there is  $x \in [q]$  where  $c_x$  has no out-neighbor in  $\{d_i : i \in [q] \setminus \{x\}\}$ . This means that the row of  $M_G[\{c_i : i \in [q]\} \rightarrow \{d_i : i \in [q]\}]$  indexed by  $c_x$  does not affect on the sum on columns indexed by  $\{d_i : i \in [q] \setminus \{x\}\}$ . Therefore, the sum of  $M_G[\{c_j\} \rightarrow \{d_i : i \in [q] \setminus \{x\}\}]$  over  $j \in [q] \setminus \{x\}$  must be a zero vector. This contradicts the minimality of  $I$ .

We deduce that there exists a sequence  $(i_1, i_2, \dots, i_y)$  of at least two distinct elements in  $[q]$  with  $i_{y+1} = i_1$  such that for each  $j \in [y]$ , there exists an edge from  $c_{i_j}$  to  $d_{i_{j+1}}$ . For each  $j \in [q]$ , let  $\ell_{i_j}$  be the length of the subpath of  $P_{i_j}$  from  $a_{i_j}$  to  $c_{i_j}$ . Observe that if  $\ell_{i_j} \leq \ell_{i_{j+1}}$ , then there is a directed path of length at most  $r$  from  $a_{i_j}$  to  $b_{i_{j+1}}$  by using the sub-path  $a_{i_j}$  to  $c_{i_j}$ , then the edge to  $d_{i_{j+1}}$  and then the sub-path to  $b_{i_{j+1}}$ , which contradicts the assumption that there is no edge from  $a_{i_j}$  to  $b_{i_{j+1}}$  in  $G^r$ . On the other hand, because of the cycle structure it is not possible that for all  $j \in [x]$ ,  $\ell_{i_j} > \ell_{i_{j+1}}$ . So, we have a contradiction.

Note that the same argument holds for undirected graphs; if an undirected graph  $G$  has rank-width  $w$ , then any power of  $G$  has mim-width at most  $w$ . This extends the two arguments in [28] that any power of an undirected graph of tree-width  $w - 1$  has mim-width at most  $w$ , and any power of an undirected graph of clique-width  $w$  has mim-width at most  $w$ , because such graphs have rank-width at most  $w$  [36, 37].

Next, we show that the  $r$ -th power of a digraph of bi-mim-width  $w$  has bi-mim-width at most  $rw$ . This will be used to prove Theorem 2.

**Lemma 3.9.** *Let  $r$  and  $w$  be positive integers. If  $(T, \mathcal{L})$  is branch-decomposition of a digraph  $G$  of bi-mim-width  $w$ , then it is a branch-decomposition of  $G^r$  has bi-mim-width at most  $rw$ .*

*Proof.* It is sufficient to prove that for every ordered vertex partition  $(A, B)$  of  $G$ , we have  $\nu(G^r[A \rightarrow B]) \leq r\nu(G[A \rightarrow B])$ . Assume  $\nu(G[A \rightarrow B]) = t$  and suppose for contradiction that  $\nu(G^r[A \rightarrow B]) \geq rt + 1$ .

Let  $\{(a_i, b_i) : i \in [rt + 1]\}$  be an induced matching of  $G^r[A \rightarrow B]$  with  $\{a_i : i \in [rt + 1]\} \subseteq A$ . For each  $i \in [rt + 1]$ , let  $P_i$  be a directed path of length at most  $r$  from  $a_i$  to  $b_i$  in  $G$ . We choose an edge  $(c_i, d_i)$  in each  $P_i$  where  $c_i \in A$  and  $d_i \in B$ . For each  $i \in [rt + 1]$ , let  $\ell_i$  be the length of the subpath of  $P_i$  from  $a_i$  to  $c_i$ . Observe that  $0 \leq \ell_i \leq r - 1$ .

By the pigeonhole principle, there exists a subset  $I$  of  $[rt + 1]$  of size at least  $t + 1$  such that for all  $i_1, i_2 \in I$ ,  $\ell_{i_1} = \ell_{i_2}$ . Since  $\nu(G[A \rightarrow B]) = t$ , there exist distinct integers  $i_1, i_2 \in I$  such that there is an edge from  $c_{i_1}$  to  $d_{i_2}$ . Then there is a path of length at most  $d$  from  $a_{i_1}$  to  $b_{i_2}$ , contradicting the assumption that there is no edge from  $a_{i_1}$  to  $b_{i_2}$  in  $G^r$ .  $\square$

## 4 Classes of digraphs of bounded bi-mim-width

In this section, we present several digraph classes of bounded bi-mim-width, which are reflexive  $H$ -graphs (Proposition 4.3), adjusted permutation digraphs (Proposition 4.7), adjusted rooted directed path digraphs (Proposition 4.9), and nice  $H$ -convex graphs (Proposition 4.11).

We recall that a digraph  $G$  is an *intersection digraph* if there exists a family  $\{(S_v, T_v) : v \in V(G)\}$  of ordered pairs of sets, called a *representation*, such that there is an edge from  $v$  to  $w$  in  $G$  if and only if  $S_v$  intersects  $T_w$ .

### 4.1 $H$ -digraphs and interval digraphs

We define  $H$ -digraphs, which generalize interval digraphs.

**Definition 4.1 ( $H$ -digraph).** Let  $H$  be an undirected graph. A digraph  $G$  is an  $H$ -digraph if there is a subdivision of  $H$  and a family  $\{(S_v, T_v) : v \in V(G)\}$  of ordered pairs of connected subgraphs of  $F$  such that  $G$  is the intersection digraph with representation  $\{(S_v, T_v) : v \in V(G)\}$ .

**Definition 4.2 (Interval digraph).** A digraph  $G$  is an interval digraph if it is a  $P_2$ -digraph.

We show that for fixed  $H$ , reflexive  $H$ -digraphs have bounded linear bi-mim-width.

**Proposition 4.3.** *Let  $H$  be an undirected graph. Given a representation of a reflexive  $H$ -digraph  $G$ , one can construct in polynomial time a linear branch decomposition of  $G$  of bi-mim-width at most  $12|E(H)|$ .*

*Proof.* Let  $m := |E(H)|$ . We may assume that  $H$  is connected. If  $H$  has no edge, then it is trivial. Thus, we may assume that  $m \geq 1$ .

Let  $G$  be a reflexive  $H$ -digraph, let  $F$  be a subdivision of  $H$ , and let  $\mathcal{M} := \{(S_v, T_v) : v \in V(G)\}$  be a given reflexive  $H$ -digraph representation of  $G$  with underlying graph  $F$ . For each  $v \in V(G)$ , choose a vertex  $\alpha_v$  in  $S_v \cap T_v$ . We may assume that vertices in  $(\alpha_v : v \in V(G))$  are pairwise distinct and they are not branching vertices, by subdividing  $F$  more and changing  $\mathcal{M}$  accordingly, if necessary.

We fix a branching vertex  $r$  of  $F$  and obtain a BFS ordering of  $F$  starting from  $r$ . We denote by  $v <_B w$  if  $v$  appears before  $w$  in the BFS ordering. We give a linear ordering  $L$  of  $G$  such that for all  $v, w \in V(G)$ , if  $\alpha_v <_B \alpha_w$ , then  $v$  appears before  $w$  in  $L$ . This can be done in linear time.

We claim that  $L$  has width at most  $12m$ . We choose a vertex  $v$  of  $G$  arbitrarily, and let  $A$  be the set of vertices in  $G$  that are  $v$  or a vertex appearing before  $v$  in  $L$ , and let  $B := V(G) \setminus A$ . It suffices to show  $\text{bimim}_G(A) \leq 12m$ . Let  $A^*$  be the set of vertices of  $F$  that are  $\alpha_v$  or a vertex appearing before  $\alpha_v$ , and let  $B^* := V(F) \setminus A^*$ . Let  $\mathcal{P}$  be the set of paths in  $F$  such that

- for every  $P \in \mathcal{P}$ ,  $P$  is a subpath of some branching path of  $F$  and it is a maximal path contained in one of  $A^*$  and  $B^*$ ,
- $\bigcup_{P \in \mathcal{P}} V(P) = V(F)$ .

Because of the property of a BFS ordering, it is easy to see that each branching path of  $F$  is partitioned into at most 3 vertex-disjoint paths in  $\mathcal{P}$ . Thus, we have  $|\mathcal{P}| \leq 3m$ . Note that two paths in  $\mathcal{P}$  from two distinct branching paths may share an endpoint.

We first show that  $\text{mim}_G^+(A) \leq 6m$ . Suppose for contradiction that  $G[A \rightarrow B]$  contains an induced matching  $M$  of size  $6m + 1$ . By the pigeonhole principle, there is a subset  $M_1 = \{(x_i, y_i) :$

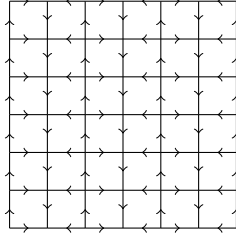


Figure 2: The interval digraph  $G_7$  in Proposition 4.5. The left bottom vertex is  $v_{1,1}$  and its right-hand vertex is  $v_{2,1}$  and so on.

$i \in [3]\}$  of  $M$  of size 3 and a path  $P$  in  $\mathcal{P}$  such that for every  $(x, y) \in M_1$ ,  $S_x$  and  $T_y$  meet on  $P$ . Let  $p_1, p_2$  be the endpoints of  $P$ .

Observe that  $V(P) \subseteq A^*$  or  $V(P) \subseteq B^*$ . So, for each  $i \in [3]$ , it is not possible that  $\alpha_{x_i}$  and  $\alpha_{y_i}$  are both contained in  $V(P)$ . It implies that each connected component of  $(S_{x_i} \cup T_{y_i}) \cap P$  contains an endpoint of  $P$ , as  $S_{x_i} \cup T_{y_i}$  is connected. Therefore, there are at least two integers  $j_1, j_2 \in [3]$  and a connected component  $C_1$  of  $(S_{x_{j_1}} \cup T_{y_{j_1}}) \cap P$  and a connected component  $C_2$  of  $(S_{x_{j_2}} \cup T_{y_{j_2}}) \cap P$  so that

- $C_1$  and  $C_2$  contain the same endpoint of  $P$ , and
- for each  $i \in [2]$ ,  $C_i$  contains a vertex of  $S_{x_{j_i}}$  and a vertex of  $T_{y_{j_i}}$ .

However, it implies that  $(x_{j_1}, y_{j_2})$  or  $(x_{j_2}, y_{j_1})$  is an edge, a contradiction.

We deduce that  $\text{mim}_G^+(A) \leq 6m$ . By a symmetric argument, we get  $\text{mim}_G^-(A) \leq 6m$ . Therefore, we have  $\text{bimim}_G(A) \leq 12m$ , as required.  $\square$

We obtain a better bound for reflexive interval digraphs.

**Proposition 4.4.** *Given a reflexive interval digraph, one can output a linear branch decomposition of bi-mim-width at most 2 in polynomial time.*

*Proof.* Let  $G$  be a given reflexive interval digraph. By Müller's recognition algorithm for interval digraphs [34], one can out its representation in polynomial time.

Now, we follow the proof of Proposition 4.3. In this case,  $\mathcal{P}$  consists of exactly two paths, one induced by  $A^*$  and the other induced by  $B^*$ . Because of it, it is not difficult to observe that  $\text{mim}_G^+(A) \leq 1$  and  $\text{mim}_G^-(A) \leq 1$  (similar to interval graphs). Thus, it has linear bi-mim-width at most 2.  $\square$

**Proposition 4.5.** *Interval digraphs have unbounded bi-mim-width.*

*Proof.* We will construct some orientation of the  $(n \times n)$ -grid as an interval digraph.

For  $i, j \in [n]$ , we construct  $S_{v_{i,j}}$  and  $T_{v_{i,j}}$  as follows. For every odd integer  $i$ , we set

- $S_{v_{i,j}} := [2(n+1)j + 2i - 1, 2(n+1)j + 2i + 1]$  and,
- $T_{v_{i,j}} := [2(n+1)(j-1) + 2i, 2(n+1)(j-1) + 2i + 1]$ ,

and for every even integer  $i$ , we set

- $S_{v_{i,j}} := [2(n+1)(j-1) + 2i, 2(n+1)(j-1) + 2i + 1]$  and

- $T_{v_{i,j}} := [2(n+1)j + 2i - 1, 2(n+1)(j-1) + 2i + 1]$ .

Let  $G_n$  be the intersection digraph on the vertex set  $\{v_{i,j} : i, j \in [n]\}$  with representation  $\{(S_{v_{i,j}}, T_{v_{i,j}}) : i, j \in [n]\}$ . Observe that for  $i, j \in [n-1]$ , if  $i$  is odd, then  $(v_{i,j}, v_{i+1,j})$  and  $(v_{i,j}, v_{i,j+1})$  are edges, and if  $i$  is even, then  $(v_{i+1,j}, v_{i,j})$  and  $(v_{i,j+1}, v_{i,j})$  are edges. See Figure 2 for an illustration.

This is an orientation of the  $(n \times n)$ -grid. It is known that the  $(n \times n)$ -grid has mim-width at least  $n/3$ ; see [47, Theorem 4.3.10]. By Lemma 3.5, it has bi-mim-width at least  $n/3$ .  $\square$

## 4.2 Permutation digraphs

Permutation digraphs are directed analogues of permutation graphs.

**Definition 4.6 (Permutation digraph).** A digraph  $G$  is a *permutation digraph* if there is a family  $\{(S_v, T_v) : v \in V(G)\}$  of line segments whose endpoints lie on two parallel lines  $\Lambda_1$  and  $\Lambda_2$  where  $G$  is the intersection digraph with representation  $\{(S_v, T_v) : v \in V(G)\}$ . A permutation digraph is *adjusted* if  $S_v$  and  $T_v$  have a common endpoint in  $\Lambda_1$  for all  $v \in V(G)$ .

We observe that interval digraphs are permutation digraphs. When we have a representation  $\{(S_v, T_v) : v \in V(G)\}$  for an interval digraph on a line  $\Lambda = \{x : x \in \mathbb{R}\}$ , we consider two copies  $\Lambda_1 = \{(x, 0) : x \in \mathbb{R}\}$  and  $\Lambda_2 = \{(x, 1) : x \in \mathbb{R}\}$  of  $\Lambda$ , and then for each  $S_v = [a_v, b_v]$ , we make  $S_v^*$  as the line segment linking  $(a_v, 0)$  and  $(b_v, 1)$ , and for each  $T_v = [c_v, d_v]$ , we make  $T_v^*$  as the line segment linking  $(c_v, 1)$  and  $(d_v, 0)$ . One can verify that  $\{(S_v^*, T_v^*) : v \in V(G)\}$  is a representation of the same digraph. Because of this and by Proposition 4.5, permutation digraphs also have unbounded bi-mim-width.

We show that adjusted permutation digraphs have linear bi-mim-width at most 4. Note that an adjusted permutation graph is reflexive, but we were not able to show that all reflexive permutation digraphs have bounded bi-mim-width. It remains open whether there is a constant bound on (linear) bi-mim-width of reflexive permutation digraphs.

**Proposition 4.7.** *Given a representation of an adjusted permutation digraph  $G$ , one can construct in polynomial time a linear branch decomposition of  $G$  of bi-mim-width at most 4.*

*Proof.* Let  $\Lambda_1 := \{(x, 0) : x \in \mathbb{R}\}$  and  $\Lambda_2 := \{(x, 1) : x \in \mathbb{R}\}$  be two lines. Let  $G$  be a given adjusted permutation digraph with its representation  $\{(S_v, T_v) : v \in V(G)\}$  where  $S_v$  and  $T_v$  are line segments whose endpoints lie on  $\Lambda_1$  and  $\Lambda_2$  and they have a common endpoint in  $\Lambda_1$ , say  $(\alpha_v, 0)$ . For each  $v \in V(G)$ , let  $(\beta_v, 1)$  be the endpoint of  $S_v$  in  $\Lambda_2$  and  $(\gamma_v, 1)$  be the endpoint of  $T_v$  in  $\Lambda_2$ .

We give a linear ordering  $L$  of  $G$  such that for all  $v, w \in V(G)$ , if  $\alpha_v < \alpha_w$ , then  $v$  appears before  $w$  in  $L$ . This can be done in linear time.

We claim that  $L$  has bi-mim-width at most 4. We choose a vertex  $v$  of  $G$  arbitrarily, and let  $A$  be the set of vertices in  $G$  that are  $v$  or a vertex appearing before  $v$  in  $L$ , and let  $B := V(G) \setminus A$ .

We verify that  $\text{mim}_G^+(A) \leq 2$ . Suppose for contradiction that  $G[A \rightarrow B]$  has an induced matching  $\{(v_i, w_i) : i \in [3]\}$  with  $v_1, v_2, v_3 \in A$ . Without loss of generality, we assume that  $\alpha_{v_1} \leq \alpha_{v_2} \leq \alpha_{v_3}$ . Observe that  $\alpha_{w_1}, \alpha_{w_2} > \alpha_{v_3}$  and  $\alpha_{w_3} \geq \alpha_{v_3}$ . Let  $w \in \{w_i : i \in [3]\}$  such that  $|\alpha_w - \alpha_{v_3}|$  is minimum. We distinguish the case depending on whether  $w = v_3$  or not.

(Case 1.  $\alpha_w \neq \alpha_{v_3}$ .) First assume that  $w = w_3$ . As  $\{(v_i, w_i) : i \in [3]\}$  is an induced matching, we have  $\beta_{v_1} < \gamma_{w_3}$  and  $\beta_{v_3} < \gamma_{w_1}$ . This implies that  $S_{v_1}$  and  $T_{w_1}$  do not meet, a contradiction. Now, assume that  $w \neq w_3$  and thus,  $S_{v_3}$  and  $T_w$  do not meet. Let  $v^*$  be the vertex where  $(v^*, w)$



is in the induced matching. As  $\{(v_i, w_i) : i \in [3]\}$  is an induced matching,  $\beta_{v^*} < \gamma_{w_3}$  and  $\gamma_w > \beta_{v_3}$ . Then  $S_{v^*}$  cannot meet  $T_w$ , a contradiction.

(Case 2.  $\alpha_w = \alpha_{v_3}$ .) In this case,  $w$  must be  $w_3$ . If  $\gamma_{w_3} \leq \beta_{v_3}$ , then  $S_{v_1}$  and  $T_{w_1}$  cannot meet. So,  $\gamma_{w_3} > \beta_{v_3}$ , and it is not difficult to verify that  $\beta_{v_3} < \beta_{v_1}, \beta_{v_2}, \gamma_{w_1}, \gamma_{w_2} < \gamma_{w_3}$ , otherwise,  $\{(v_i, w_i) : i \in [3]\}$  cannot be an induced matching. If  $\beta_{v_1} \leq \beta_{v_2}$ , then  $T_{w_1}$  has to meet  $S_{v_2}$ , a contradiction. Similarly, if  $\beta_{v_2} < \beta_{v_1}$ , then  $T_{w_2}$  has to meet  $S_{v_1}$ , a contradiction.

It shows that  $\text{mim}_G^+(A) \leq 2$ , as claimed. By a symmetric argument, we have  $\text{mim}_G^-(A) \leq 2$  and  $\text{bimim}_G(A) \leq 4$ .

We conclude that  $L$  has bi-mim-width at most 4.  $\square$

### 4.3 Rooted directed path digraphs

A *directed tree* is an orientation of a tree, and it is *rooted* if there is a root node  $r$  such that every vertex is reachable from  $r$  by a directed path. Gavril [24] introduced the class of *rooted directed path graphs*, that are intersection graphs of directed paths in a rooted directed tree. We introduce its directed analogue.

**Definition 4.8 (Rooted directed path digraph).** A digraph  $G$  is a *rooted directed path digraph* if there is a rooted directed tree  $T$  and a family  $\{(S_v, T_v) : v \in V(G)\}$  of pairs of directed paths in  $T$  where  $G$  is the intersection digraph with representation  $\{(S_v, T_v) : v \in V(G)\}$ . A directed rooted path digraph is *adjusted* if for every  $v \in V(G)$ , the endpoint of  $S_v$  that is farther from the root is the same as the endpoint of  $T_v$  that is farther from the root.

Clearly, interval digraphs are rooted directed path digraphs, and therefore, rooted directed path digraphs have unbounded bi-mim-width. We prove that adjusted rooted directed path digraphs have bounded bi-mim-width and have unbounded linear bi-mim-width.

**Proposition 4.9.** *Given a representation of an adjusted rooted directed path digraph  $G$ , one can construct in polynomial time a branch decomposition of  $G$  of bi-mim-width at most 2. Adjusted rooted directed path digraphs have unbounded linear bi-mim-width.*

*Proof.* Let  $F$  be a rooted directed tree with root node  $r$ . Let  $G$  be an adjusted rooted directed path digraph with representation  $\mathcal{M} = \{(S_v, T_v) : v \in V(G)\}$  where  $S_v$  and  $T_v$  are directed paths in  $F$ .

Now, we modify  $(F, \mathcal{M})$  into  $(F^*, \mathcal{M}^* = \{(S_v^*, T_v^*) : v \in V(G)\})$  so that  $F^*$  is a rooted directed tree and  $\mathcal{M}^*$  is an adjusted rooted directed path digraph representation of  $G$  consisting of directed paths in  $F^*$  with additional conditions that

- every internal node of  $F^*$  has out-degree at most 2,
- for every vertex  $v \in V(G)$ , the endpoint  $\alpha_v$  of  $S_v^*$  that is farther from the root of  $F^*$  has out-degree at most 1 in  $F^*$ , and nodes in  $(\alpha_v : v \in V(G))$  are pairwise distinct.

Let  $(F_1, \mathcal{M}_1) := (F, \mathcal{M})$ . For every  $v \in V(G)$ , let  $\alpha_v^1$  be the endpoint of  $S_v$  that is farther from the root in  $F$ . We recursively construct  $(F_i, \mathcal{M}_i = \{(S_v^i, T_v^i) : v \in V(G)\}, (\alpha_v^i : v \in V(G)))$  until we get the desired conditions.

- Assume that  $F_i$  has a node  $t$  of out-degree at least 3. Let  $t_1, \dots, t_x$  be the out-neighbors of  $t$  in  $F_i$ . Now, we remove all edges between  $t$  and  $\{t_1, \dots, t_x\}$ , and then add a directed path

$p_1 p_2 \cdots p_x$  and add an edge  $(t, p_1)$  and edges  $(p_i, t_i)$  for all  $i \in [x]$ . Now, for every path  $P$  containing  $tt_j$  for some  $j$  in the representation  $\mathcal{M}_i$  we replace  $tt_j$  with a path  $tp_1 p_2 \cdots p_j t_j$ . The other paths do not change. The resulting rooted directed tree and representation are  $F_{i+1}$  and  $\mathcal{M}_{i+1}$ , respectively.

- Assume that  $\alpha_v^i = \alpha_w^i = t$  for some distinct vertices  $v$  and  $w$  in  $G$ . Let  $q$  be an out-neighbor of  $t$  in  $F_i$  if one exists, and otherwise we attach a new node  $q$  and add an edge  $(t, q)$ . We replace  $(t, q)$  with a directed path  $tt'q$  to obtain  $F_{i+1}$ . For every path in  $\mathcal{M}_i$  containing  $(t, q)$  we replace  $(t, q)$  with a directed path  $tt'q$ , and assign  $\alpha_v^{i+1} := t'$ . For every path in  $\{S_v^i, T_v^i\}$ , we extend it by adding  $tt'$ . The other paths do not change. The resulting representation is  $\mathcal{M}_{i+1}$ .
- Assume that  $\alpha_v^i = t$  for some vertex  $v$  in  $G$ , where  $t$  is a node of out-degree at least 2. Let  $q$  be an out-neighbor of  $t$  in  $F_i$ . We replace  $(t, q)$  with a directed path  $tt'q$  to obtain  $F_{i+1}$ . For every path in  $\{S_v^i, T_v^i\}$ , we extend it by adding  $tt'$ , and we assign  $\alpha_v^{i+1} := t'$ . The other paths do not change. The resulting representation is  $\mathcal{M}_{i+1}$ .

In each iteration, either the number of nodes of out-degree at least 3 decreases, or the number of pairs of vertices  $v$  and  $w$  for which  $\alpha_v^i = \alpha_w^i$  decreases, or the number of vertices  $v$  for which  $\alpha_v^i$  is a node of out-degree at least 3 decreases. Also, the process in each iteration preserve that the representation is an adjusted rooted directed path digraph representation. Thus, at the end, we obtain an adjusted rooted directed path digraph representation  $(F^*, \mathcal{M}^*)$  with the set of common endpoints  $(\alpha_v^* : v \in V(G))$ , as desired. Let  $U$  be the underlying undirected tree of  $F^*$ .

For each  $v \in V(G)$ , we add a new node  $\beta_v$  and add an edge  $\beta_v \alpha_v^*$  to  $U$ , and obtain a tree  $U^*$ . Now, let  $e$  be an edge of  $U^*$ , and let  $U_X$  and  $U_Y$  be the connected component of  $U^* - e$  where  $U_Y$  contains the root of  $F^*$ . Let  $(X_e, Y_e)$  be the cut of  $G$  where  $X_e$  is the set of vertices  $v$  for which  $\alpha_v^*$  is in  $U_X$ , and  $Y_e = V(G) \setminus X_e$ .

We claim that  $\text{mim}_G^+(X_e) \leq 1$ . Suppose for contradiction that there is an induced matching  $\{(v_i, w_i) : i \in [2]\}$  with  $v_1, v_2 \in X_e$ . Let  $Q$  be the path in  $U^*$  from the root to  $e$ .

Assume that the distance from  $e$  to  $T_{w_1}$  in  $U^*$  is at most the distance from  $e$  to  $T_{w_2}$ . Since  $S_{v_1}$  and  $S_{v_2}$  are directed paths,  $T_{w_1}$  and  $T_{w_2}$  contain a vertex of  $Q$ . It means that  $S_{v_2}$  has to meet  $T_{w_1}$ , which is a contradiction. If the distance from  $e$  to  $T_{w_2}$  in  $U^*$  is at most the distance from  $e$  to  $T_{w_1}$ , then  $S_{v_1}$  has to meet  $T_{w_2}$ , a contradiction.

Thus, we have  $\text{mim}_G^+(X_e) \leq 1$ , and by symmetry, we can show that  $\text{mim}_G^-(X_e) \leq 1$ . It implies that  $\text{bimim}_G(X_e) \leq 2$ . By smoothing degree 2 nodes of  $U^*$ , we obtain a branch-decomposition of bi-mim-width at most 2.

Now, we argue that adjusted rooted directed path digraphs have unbounded linear bi-mim-width. It is well known that trees are rooted directed path graphs. Also, Høgemo, Telle, and Vågset [27] proved that trees have unbounded linear mim-width. As trees are rooted directed path graphs, we can obtain the biorientations of trees as adjusted rooted directed path digraphs, where  $S_v = T_v$  for all  $v \in V(G)$ . As trees have unbounded linear mim-width, adjusted rooted directed path digraphs have unbounded linear bi-mim-width.  $\square$

#### 4.4 $H$ -convex digraphs

As a generalization of convex graphs, Brettell, Munaro, and Paulusma [12] introduced  $H$ -convex graphs. Note that they defined  $\mathcal{H}$ -convex graphs, rather than  $H$ -convex graphs, where  $\mathcal{H}$  is a family of graphs. However, they mostly considered  $\mathcal{H}$  as the set of all subdivisions of a fixed graph  $H$ ,

so it seems better to call them simply  $H$ -convex graphs. We generalize this notion to  $H$ -convex digraphs.

**Definition 4.10 ( $H$ -convex digraph).** Let  $H$  be an undirected graph. A bipartite digraph  $G$  with bipartition  $(A, B)$  is an  $H$ -convex digraph if there is a subdivision  $F$  of  $H$  with  $V(F) = A$  such that for every vertex  $b$  of  $B$ , each of  $N_G^+(v)$  and  $N_G^-(v)$  induces a connected subgraph of  $F$ . An  $H$ -convex digraph is *nice* if for every vertex  $b$  of  $B$ , there is a bi-directed edge between  $b$  and some vertex of  $A$ .

In principle,  $H$ -convex digraphs and  $H$ -digraphs are closely related, as  $\{(N_G^+(v), N_G^-(v)) : v \in B\}$  can be seen as an  $H$ -digraph representation on  $F$ . We prove that nice  $H$ -convex digraphs have linear bi-mim-width at most  $12|E(H)|$ .

**Proposition 4.11.** *Let  $H$  be an undirected graph. Given a nice  $H$ -convex digraph  $G$  with its bipartition  $(A, B)$ , one can construct in polynomial time a linear branch decomposition of  $G$  of bi-mim-width at most  $12|E(H)|$ .*

*Proof.* Let  $m := |E(H)|$ . Let  $G$  be a reflexive  $H$ -convex digraph with bipartition  $(X, Y)$ . Let  $F$  be a subdivision of  $H$  with  $X = V(F)$  and let  $\mathcal{M} = \{(N_G^+(y), N_G^-(y)) : y \in Y\}$ . For each  $y \in Y$ , we choose  $\alpha_y \in N_G^+(y) \cap N_G^-(y)$ . By adding more vertices in  $X$  if necessary (corresponding to a subdivision of  $F$ ), we may assume that vertices in  $(\alpha_y : y \in Y)$  are pairwise distinct and they are not branching vertices of  $F$ . This assumption is possible because of Lemma 3.3.

We fix a vertex  $r$  of  $F$  and obtain a BFS ordering of  $F$  starting from  $r$ . We denote by  $v <_B w$  if  $v$  appears before  $w$  in the BFS ordering. This gives an ordering of  $V(F) = X$ . Now, we extend it to an ordering  $L$  of  $V(G)$  by, for each  $y \in Y$ , putting  $y$  just after  $\alpha_y$ .

We claim that  $L$  has width at most  $12m$ . We choose a vertex  $v$  of  $G$  arbitrarily, and let  $A$  be the set of vertices in  $G$  that are  $v$  or a vertex appearing before  $v$  in  $L$ , and let  $B := V(G) \setminus A$ . It is sufficient to show that  $\text{mim}_G^+(A) \leq 6m$ . By symmetry, we will get  $\text{mim}_G^-(A) \leq 6m$ .

Let  $A_X := A \cap X$ ,  $A_Y := A \cap Y$ ,  $B_X := B \cap X$ , and  $B_Y := B \cap Y$ . We first show that  $\nu(G[A_X \rightarrow B_Y]) \leq 4m$ .

Suppose for contradiction that  $G[A_X \rightarrow B_Y]$  contains an induced matching  $M$  of size  $4m + 1$ . As defined in Proposition 4.3, we define  $\mathcal{P}$  as the set of paths in  $F$  such that

- for every  $P \in \mathcal{P}$ ,  $P$  is a subpath of some branching path of  $F$  and it is a maximal path contained in one of  $A_X$  and  $B_X$ ,
- $\bigcup_{P \in \mathcal{P}} V(P) = V(F)$ .

Note that each branching path contains at most two paths in  $\mathcal{P}$  that are contained in  $A_X$ . By the pigeonhole principle, there is a subset  $M_1 = \{(x_i, y_i) : i \in [3]\}$  of  $M$  of size 3 and a path  $P \in \mathcal{P}$  with  $V(P) \subseteq A_X$  such that for every  $(x, y) \in M_1$ ,  $x$  is in  $P$ . Observe that  $\alpha_{y_i}$  is not in  $P$  for every  $i \in [3]$ .

Let  $p_1$  and  $p_2$  be the endpoints of  $P$ . Observe that for each  $i \in [3]$ ,  $N_G^-(y_i)$  contains either the subpath of  $P$  from  $p_1$  to  $x_i$  or the subpath of  $P$  from  $x_1$  to  $p_2$ . Without loss of generality, we may assume that for each  $i \in [2]$ ,  $N_G^-(y_i)$  contains the subpath of  $P$  from  $p_1$  to  $x_i$ . But this implies that  $(x_1, y_2)$  or  $(x_2, y_1)$  is an edge, contradicting the assumption that  $M$  is an induced matching.

We deduce that  $\nu(G[A_X \rightarrow B_Y]) \leq 4m$ . Note that each branching path contains at most one path in  $\mathcal{P}$  that is contained in  $B_x$ . Thus, by a similar argument, we can show that  $\nu(G[A_Y \rightarrow B_X]) \leq 2m$ . So, we have  $\text{mim}_G^+(A) = \nu(G[A \rightarrow B]) \leq 6m$ . By symmetry, we get  $\text{mim}_G^-(A) \leq 6m$  as well, and these imply that  $\text{bimim}_G(A) \leq 12m$ , as required.  $\square$

**Proposition 4.12.**  *$P_2$ -convex digraphs have unbounded bi-mim-width.*

*Proof.* We recall the interval digraph representation of an orientation of the  $(n \times n)$ -grid, given in Proposition 4.5. For  $i, j \in [n]$ , we construct  $S_{i,j}$  and  $T_{i,j}$  as follows. For every odd integer  $i$ , we set

- $S_{i,j} := [2(n+1)j + 2i - 1, 2(n+1)j + 2i + 1]$  and,
- $T_{i,j} := [2(n+1)(j-1) + 2i, 2(n+1)(j-1) + 2i + 1]$ ,

and for every even integer  $i$ , we set

- $S_{i,j} := [2(n+1)(j-1) + 2i, 2(n+1)(j-1) + 2i + 1]$  and
- $T_{i,j} := [2(n+1)j + 2i - 1, 2(n+1)(j-1) + 2i + 1]$ .

Now, we create a bipartite digraph  $G_n$  with bipartition  $(A, B = \{v_{i,j} : i, j \in [n]\})$  such that for  $i, j \in [n]$ ,

- if  $i$  is odd, then  $N_G^+(v_{i,j}) = S_{i,j}$  and  $N_G^-(v_{i,j}) = T_{i,j}$ ,
- if  $i$  is even, then  $N_G^+(v_{i,j}) = T_{i,j}$  and  $N_G^-(v_{i,j}) = S_{i,j}$ .

It is not difficult to verify that  $G_n$  contains an orientation of the 1-subdivision of the  $(n \times n)$ -grid as an induced subdigraph. By modifying the proof for the fact that the  $(n \times n)$ -grid has mim-width at least  $n/3$ , it is straightforward to show that the 1-subdivision of the  $(n \times n)$ -grid has mim-width at least  $n/3$ . Therefore, by Lemma 3.5,  $G_n$  has bi-mim-width at least  $n/3$ .  $\square$

## 5 Algorithmic applications

In this section we give the algorithmic applications of the width measure bi-mim-width. In particular, we show that all directed locally checkable vertex subset and all directed locally checkable vertex partitioning problems can be solved in XP time parameterized by the bi-mim-width of a given branch decomposition of the input digraph. We do so by adapting the framework of the  $d$ -neighborhood equivalence relation introduced by Bui-Xuan et al. [14] to digraphs. For an undirected graph  $G$ , given a set  $A \subseteq V(G)$ , two subsets  $X$  and  $Y$  of  $A$  are  $d$ -neighborhood equivalent w.r.t.  $A$  if the intersection of the neighborhood of each vertex in  $\overline{A}$  with  $X$  and  $Y$  have the same size, when counting up to  $d$ .

In the adaptation of this concept to digraphs, we essentially take the cartesian product of the  $d$ -neighborhood equivalences given by the edges going from  $A$  to  $\overline{A}$  and the edges going from  $\overline{A}$  to  $A$ . In the resulting  $d$ -bi-neighborhood equivalence relation of a set  $A$  of vertices in some digraph  $G$ , two subsets  $X$  and  $Y$  of  $A$  are  $d$ -bi-neighborhood equivalent, if for each vertex of  $\overline{A}$ , the number of both its in-neighbors in  $X$  and the number of its out-neighbors in  $X$  are equal to the number of its in- and out-neighbors in  $Y$ , respectively, when counted up to  $d$ . We show that this notion allows us to lift the frameworks presented in [14] to the realm of digraphs and prove the aforementioned results.

The rest of this section is organized as follows. In Subsection 5.1, we formally define the  $d$ -bi-neighborhood equivalence relation and show how to efficiently compute descriptions of its equivalence classes. In Subsection 5.2 we give the algorithms for generalized directed domination problems and in Subsection 5.3 for the directed vertex partitioning problems. We discuss how to use these algorithms to solve distance- $r$  versions of DLCV problems in Subsection 5.4.

## 5.1 $d$ -Bi-neighborhood-equivalence

We now present the central notion that is used in our algorithms, the  $d$ -bi-neighborhood equivalence relation which we introduced informally earlier. The reason why it compares sizes of the intersection of a neighborhood with subsets only up to some fixed integer  $d$  is as follows. The subsets of natural numbers that characterize locally checkable vertex subset/partitioning problems can be fully characterized when counting in- and out-neighbors up to some constant  $d$ , depending on the described problem. Therefore, if a vertex  $v$  has more than  $d$  for instance out-neighbors in two sets  $X$  and  $Y$ , then these two sets look the same to  $v$  in terms of its out-neighborhood.

In the following definition, we present the  $d$ -in-neighborhood equivalence relation and the  $d$ -out-neighborhood equivalence relation separated before combining them to the  $d$ -bi-neighborhood equivalence relation, since in some proofs it is convenient to only consider the edges going in one direction.

**Definition 5.1.** Let  $d \in \mathbb{N}$ . Let  $G$  be a digraph and  $A \subseteq V(G)$ . For two sets  $X, Y \subseteq A$ , we say that  $X$  and  $Y$  are  $d$ -out-neighborhood equivalent, written  $X \equiv_{d,A}^+ Y$ , if<sup>1</sup>

$$\forall u \in V(G) \setminus A: \min\{d, |N^-(u) \cap X|\} = \min\{d, |N^-(u) \cap Y|\}.$$

Similarly, we say that  $X$  and  $Y$  are  $d$ -in-neighborhood equivalent, written  $X \equiv_{d,A}^- Y$ , if

$$\forall u \in V(G) \setminus A: \min\{d, |N^+(u) \cap X|\} = \min\{d, |N^+(u) \cap Y|\}.$$

If  $X \equiv_{d,A}^+ Y$  and  $X \equiv_{d,A}^- Y$  then we say that  $X$  and  $Y$  are  $d$ -bi-neighborhood equivalent and write  $X \equiv_{d,A}^\pm Y$ .

The runtime of the algorithms in this section crucially depend on the number of equivalence classes of the  $d$ -bi-neighborhood equivalence relations associated with cuts induced by a branch decomposition of the input graph. For  $d, G$ , and  $A$  as in the previous definition, we denote by  $\text{nec}(\equiv_{d,A}^\pm)$  the number of equivalence classes of  $\equiv_{d,A}^\pm$ . If  $(T, \mathcal{L})$  is a branch decomposition of  $G$ , we let

$$\text{nec}_d(T, \mathcal{L}) = \max_{t \in V(T)} \max\{\text{nec}(\equiv_{d,V_t}^\pm), \text{nec}(\equiv_{d,\overline{V_t}}^\pm)\}.$$

### Descriptions of equivalence classes of $\equiv_{d,A}^\pm$

Since  $\equiv_{d,A}^\pm$  is an equivalence relation over subsets of  $A$ , we cannot trivially enumerate all its equivalence classes without risking an exponential running time. We now show that we can enumerate the equivalence classes with a relatively small overhead depending polynomially on  $n$ ,  $d$ , and  $\log(\text{nec}(\equiv_{d,A}^\pm))$ . This enumeration is based on pairs of vectors called  $d$ -bi-neighborhoods of a subset  $X$  of  $A$ , one that describes the in-neighborhood of vertices in  $\overline{A}$  intersected with  $X$ , and one for the out-neighborhood.

**Definition 5.2.** Let  $G$  be a digraph,  $X \subseteq A \subseteq V(G)$ , and  $d \in \mathbb{N}$ . The  $d$ -out-neighborhood of  $X$ , denoted by  $U_{d,A}^+(X)$  is a vector in  $\{0, 1, \dots, d\}^{\overline{A}}$ , which stores for every vertex  $v \in \overline{A}$  the minimum between  $d$  and the number of in-neighbors of  $v$  in  $X$ . Formally,

$$U_{d,A}^+(X) = (\min\{d, |N^-(v) \cap X|\})_{v \in \overline{A}}.$$

---

<sup>1</sup>Since the definition is given in terms of vertices from  $\overline{A}$ , we consider the directions of the edges in reverse, i.e., we consider  $N^-(v)$  for  $v \in \overline{A}$  when defining  $\equiv^+$ .

<p><b>Input</b> : Digraph <math>G</math>, <math>A \subseteq V(G)</math>, and <math>d \in \mathbb{N}</math>.</p> <p><b>Output:</b> <math>\mathcal{U}_{d,A}^\pm</math></p> <pre> 1 Let <math>\mathcal{U}</math> be an empty balanced binary search tree;   /* To enable binary search in <math>\mathcal{U}</math> we assume a fixed ordering on <math>V(G)</math>; the vectors in     <math>\mathbb{N}^{\bar{A}}</math> can then be compared lexicographically. */ 2 Let <math>\mathcal{B} = \{\emptyset\}</math>; 3 while <math>\mathcal{B} \neq \emptyset</math> do 4   Let <math>\mathcal{B}' = \emptyset</math>; 5   foreach <math>R \in \mathcal{B}</math> do 6     for <math>v \in V(G)</math> do 7       Let <math>R' = R \cup \{v\}</math>; 8       if <math>U_{d,A}^\pm(R') \notin \mathcal{U}</math> then add <math>U_{d,A}^\pm(R')</math> to <math>\mathcal{U}</math> and <math>R'</math> to <math>\mathcal{B}'</math>; 9   Let <math>\mathcal{B} = \mathcal{B}'</math>; 10 return <math>\mathcal{U}</math> </pre>
---

**Algorithm 1:** Algorithm to compute all  $d$ -bi-neighborhoods.

Similarly, the  $d$ -in-neighborhood, denoted by  $U_{d,A}^-(X)$ , is the vector

$$U_{d,A}^-(X) = (\min\{d, |N^+(v) \cap X|\})_{v \in \bar{A}}.$$

We refer to the pair  $(U_{d,A}^+(X), U_{d,A}^-(X))$  as the  $d$ -bi-neighborhood  $U_{d,A}^\pm(X)$ ; and we denote the set of all  $d$ -bi-neighborhoods as  $\mathcal{U}_{d,A}^\pm$ .

*Observation 5.3.* Let  $G$  be a digraph and  $X, Y \subseteq A \subseteq V(G)$ . Then,  $X \equiv_{d,A}^\pm Y$  if and only if  $U_{d,A}^\pm(X) = U_{d,A}^\pm(Y)$ .

By the previous observation, there is a natural bijection between the  $d$ -bi-neighborhoods and the equivalence classes of  $\equiv_{d,A}^\pm$ . In our algorithm we will therefore use the  $d$ -bi-neighborhoods as descriptions for the equivalence classes of  $\equiv_{d,A}^\pm$ . We now show that we can efficiently enumerate them. While the ideas are parallel to the algorithm presented in [14], we work directly with the  $d$ -bi-neighborhoods rather than with representatives to streamline the presentation.

**Lemma 5.4.** *Let  $G$  be a digraph on  $n$  vertices,  $A \subseteq V(G)$ , and  $d \in \mathbb{N}$ . There is an algorithm that enumerates all members of  $\mathcal{U}_{d,A}^\pm$  in time  $\mathcal{O}(\text{nec}(\equiv_{d,A}^\pm) \log \text{nec}(\equiv_{d,A}^\pm) \cdot dn^2)$ . Furthermore, for each  $Y \in \mathcal{U}_{d,A}^\pm$ , the algorithm can provide some  $X \subseteq A$  with  $U_{d,A}^\pm(X) = Y$ .*

*Proof.* We describe the procedure to enumerate  $\mathcal{U}_{d,A}^\pm$  in Algorithm 1.

Let us argue that this algorithm is correct. It is easy to observe that  $\mathcal{U}$  only contains pairwise distinct  $d$ -bi-neighborhoods. Suppose for a contradiction that there is a set  $X \subseteq A$  such that  $U_{d,A}^\pm(X) \notin \mathcal{U}$ , and assume wlog. that  $X$  is a minimal subset of vertices whose  $d$ -bi-neighborhood is not contained in  $\mathcal{U}$ . Let  $u \in X$ . We know that for all  $Y \in \mathcal{U}$ ,  $U_{d,A}^\pm(X \setminus \{u\}) \neq Y$ , for otherwise,  $U_{d,A}^\pm(X)$  would have been added to  $\mathcal{U}$ . But this contradicts the minimality of  $X$ .

Algorithm 1 can easily be modified to satisfy the second claim of the lemma: In line 8, instead of adding only  $U_{d,A}^\pm(R')$  to  $\mathcal{U}$ , we may add the pair  $(R', U_{d,A}^\pm(R'))$ .

We analyze the runtime as follows. For each  $d$ -bi-neighborhood that is added to  $\mathcal{U}$ , we test for at most  $n$  additional sets whether their  $d$ -bi-neighborhoods need to be added to  $\mathcal{U}$  or not. Together



with Observation 5.3, this implies that we test for at most  $\mathcal{O}(\text{nec}(\equiv_{d,A}^\pm) \cdot n)$  sets whether they should be added to  $\mathcal{U}$  or not. Computing the  $d$ -bi-neighborhood of a set  $X \subseteq A$  can be done in time  $\mathcal{O}(d \cdot n)$ , and since  $\mathcal{U}$  is a balanced binary tree, we can check for containment in time  $\mathcal{O}(\log(\text{nec}(\equiv_{d,A}^\pm)) \cdot dn)$ , so the total runtime of the algorithm is  $\mathcal{O}(\text{nec}(\equiv_{d,A}^\pm) \log \text{nec}(\equiv_{d,A}^\pm) \cdot dn^2)$ .  $\square$

## 5.2 Generalized Directed Domination Problems

In this section we use the  $d$ -bi-neighborhood equivalence relation to give algorithms for problems that ask for a maximum- or minimum-size set that can be expressed as a  $(\sigma^+, \sigma^-, \rho^+, \rho^-)$ -set. The algorithm is bottom-up dynamic programming along the given branch decomposition  $(T, \mathcal{L})$  of the input digraph  $G$ , which we assume to be rooted in an arbitrary degree two node. For a node  $t \in V(T)$ , we let  $V_t$  be the vertices of  $G$  that are mapped to a leaf in the subtree of  $T$  rooted at  $t$ . Before we proceed with its description, we recall the formal definition here of  $(\sigma^+, \sigma^-, \rho^+, \rho^-)$ -sets.

**Definition 5.5.** Let  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$ , and let  $\Sigma = (\sigma^+, \sigma^-)$  and  $R = (\rho^+, \rho^-)$ . Let  $G$  be a digraph and  $S \subseteq V(G)$ . We say that  $S$   $(\sigma^+, \sigma^-, \rho^+, \rho^-)$ -dominates  $G$ , or simply that  $S$   $(\Sigma, R)$ -dominates  $G$ , if:

$$\forall v \in V(G): |N^+(v) \cap S| \in \begin{cases} \sigma^+, & \text{if } v \in S \\ \rho^+, & \text{if } v \notin S \end{cases} \quad \text{and} \quad |N^-(v) \cap S| \in \begin{cases} \sigma^-, & \text{if } v \in S \\ \rho^-, & \text{if } v \notin S \end{cases}$$

For better readability, it is often convenient to gather the sets  $\sigma^+$  and  $\sigma^-$  as one and the sets  $\rho^+$  and  $\rho^-$  as one. We will mostly use the resulting  $(\Sigma, R)$ -notation. We now recall the definition of the  $d$ -value of a finite or co-finite set. Informally speaking, this value tells us how far we have to count until in order to completely describe a finite or co-finite set.

**Definition 5.6.** Let  $d(\mathbb{N}) = 0$ . For a finite or co-finite set  $\mu \subseteq \mathbb{N}$ , let

$$d(\mu) = 1 + \min\{\max_{x \in \mathbb{N}} x \in \mu, \max_{x \in \mathbb{N}} x \notin \mu\}.$$

For finite or co-finite  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$ ,  $\Sigma = (\sigma^+, \sigma^-)$  and  $R = (\rho^+, \rho^-)$

$$d(\sigma^+, \sigma^-, \rho^+, \rho^-) = d(\Sigma, R) = \max\{d(\sigma^+), d(\sigma^-), d(\rho^+), d(\rho^-)\}.$$

As our algorithm progresses, it keeps track of partial solutions that may become a  $(\Sigma, R)$ -set once the computation has finished. This does not necessarily mean that at each node  $t \in V(T)$ , such a partial solution  $X \subseteq V_t$  has to be a  $(\Sigma, R)$ -dominating set of  $G[V_t]$ . Instead, we additionally consider what is usually referred to as the “expectation from the outside” [14] in form of a subset  $Y$  of  $\overline{V_t}$  such that  $X \cup Y$  is a  $(\Sigma, R)$ -dominating set of  $G[V_t]$ . This is captured in the following definition.

**Definition 5.7.** Let  $\mu^+, \mu^- \subseteq \mathbb{N}$  and let  $M = (\mu^+, \mu^-)$ . Let  $G$  be a digraph, and let  $A \subseteq V(G)$  and  $X \subseteq V(G)$ . We say that  $X$   $M$ -dominates  $A$  if for all  $v \in A$ , we have that  $|N^+(v) \cap X| \in \mu^+$  and  $|N^-(v) \cap X| \in \mu^-$ . Let  $\Sigma$  and  $R$  be as above. For  $X \subseteq A$  and  $Y \subseteq \overline{A}$ , we say that  $(X, Y)$   $(\Sigma, R)$ -dominates  $A$ , if  $X \cup Y$   $\Sigma$ -dominates  $X$  and  $X \cup Y$   $R$ -dominates  $A \setminus X$ .

The next lemma shows that the previous definition behaves well with respect to  $\equiv_{d,A}^\pm$ .

**Lemma 5.8.** Let  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$  be finite or co-finite, let  $\Sigma = (\sigma^+, \sigma^-)$  and  $R = (\rho^+, \rho^-)$ , and let  $d = d(\Sigma, R)$ . Let  $G$  be a digraph and let  $A \subseteq V(G)$ . Let  $X \subseteq A$  and  $Y, Y' \subseteq V(G) \setminus A$  such that  $Y \equiv_{d,A} Y'$ . Then,  $(X, Y)$   $(\Sigma, R)$ -dominates  $A$  if and only if  $(X, Y')$   $(\Sigma, R)$ -dominates  $A$ .

*Proof.* Suppose  $(X, Y)$   $(\Sigma, R)$ -dominates  $A$ . Let  $v \in A \setminus X$ . Since  $Y \equiv_{d, A}^- Y'$ , we have that

$$\min\{d, |N^+(v) \cap Y|\} = \min\{d, |N^+(v) \cap Y'|\}. \quad (1)$$

If  $|N^+(v) \cap Y| \leq d$ , then immediately by (1) we have that  $|N^+(v) \cap Y| = |N^+(v) \cap Y'|$ . Therefore,  $|N^+(v) \cap (X \cup Y)| = |N^+(v) \cap (X \cup Y')| \in \rho^+$ . If  $|N^+(v) \cap Y| > d$ , then by the definition of the  $d$ -value we have that for all  $n > d$ ,  $n \in \rho^+$ . By (1), this implies that  $|N^+(v) \cap Y'| \in \rho^+$ , and in particular that  $|N^+(v) \cap (X \cup Y')| \in \rho^+$ . Similarly we can show that  $|N^-(v) \cap (X \cup Y')| \in \rho^-$ , and so  $X \cup Y'$   $R$ -dominates  $A \setminus X$ . Since  $Y \equiv_{d, A}^+ Y'$ , we can use the same arguments to show that for all  $v \in X$ ,  $|N^+(v) \cap (X \cup Y')| \in \sigma^+$  and  $|N^-(v) \cap (X \cup Y')| \in \sigma^-$ , and we conclude that  $(X, Y')$   $(\Sigma, R)$ -dominates  $A$ .

A symmetric argument yields the other direction, i.e., if  $(X, Y')$   $(\Sigma, R)$ -dominates  $A$  then  $(X, Y)$   $(\Sigma, R)$ -dominates  $A$ .  $\square$

We now turn to the definition of the table entries. To describe an equivalence class  $\mathcal{Q}$  of  $\equiv_{d, A}^\pm$  we use the  $d$ -bi-neighborhoods of its members. Note that by Observation 5.3, the following notion of a description of an equivalence class is well-defined.

**Definition 5.9.** Let  $G$  be a digraph,  $A \subseteq V(G)$ , and  $d \in \mathbb{N}$ . For an equivalence class  $\mathcal{Q}$  of  $\equiv_{d, A}^\pm$ , its *description*, denoted by  $\text{desc}(\mathcal{Q})$ , is the  $d$ -bi-neighborhood of all members of  $\mathcal{Q}$ .

As the table entries are indexed by equivalence classes of  $\equiv_{d, A}^\pm$ , we use their descriptions as compact representations.

**Definition 5.10.** Let  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$  be finite or co-finite, let  $\Sigma = (\sigma^+, \sigma^-)$ ,  $R = (\rho^+, \rho^-)$ , and  $d = d(\Sigma, R)$ . Let  $\text{opt}$  stand for  $\min$  if we consider a minimization problem and for  $\max$  if we consider a maximization problem. Let  $G$  be a digraph with branch decomposition  $(T, \mathcal{L})$  and let  $t \in V(T)$ . For an equivalence class  $\mathcal{Q}_t$  of  $\equiv_{d, V_t}^\pm$ , and an equivalence class  $\mathcal{Q}_{\bar{t}}$  of  $\equiv_{d, \bar{V}_t}^\pm$ , we let:

$$\text{Tab}_t[\text{desc}(\mathcal{Q}_t), \text{desc}(\mathcal{Q}_{\bar{t}})] = \begin{cases} \text{opt}_{S \subseteq V_t} |S|: & S \in \mathcal{Q}_t \text{ and for any } S_{\bar{t}} \in \mathcal{Q}_{\bar{t}}: \\ & (S, S_{\bar{t}}) (\Sigma, R)\text{-dominates } V_t \\ \infty & \text{if } \text{opt} = \min \text{ and no such } S \text{ exists} \\ -\infty & \text{if } \text{opt} = \max \text{ and no such } S \text{ exists} \end{cases}$$

We use the shorthand ' $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}]$ ' for ' $\text{Tab}_t[\text{desc}(\mathcal{Q}_t), \text{desc}(\mathcal{Q}_{\bar{t}})]$ '.

We first initialize the table entries for all  $t \in V(T)$  as follows. We use the algorithm of Lemma 5.4 to enumerate all descriptions of equivalence classes  $\mathcal{Q}_t$  of  $\equiv_{d, V_t}^\pm$  and  $\mathcal{Q}_{\bar{t}}$  of  $\equiv_{d, \bar{V}_t}^\pm$  and we let:

$$\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] = \begin{cases} -\infty, & \text{if } \text{opt} = \max, \text{ and} \\ \infty, & \text{if } \text{opt} = \min \end{cases} \quad (2)$$

**Leaves of  $T$ .** For a leaf  $\ell \in V(T)$ , let  $v \in V(G)$  be such that  $\mathcal{L}(v) = \ell$ . Clearly,  $\equiv_{d, \{v\}}$  has only two equivalence classes, namely the one containing  $\emptyset$  and the one containing  $\{v\}$ . For each equivalence class  $\mathcal{Q}$  of  $\equiv_{d, V(G) \setminus \{v\}}$ , let  $R \in \mathcal{Q}$  which we can assume is given to us by Lemma 5.4.

- If  $|N^+(v) \cap R| \in \sigma^+$  and  $|N^-(v) \cap R| \in \sigma^-$ , then  $\text{Tab}_\ell[\{\{v\}\}, \mathcal{Q}] = 1$ .
- If  $|N^+(v) \cap R| \in \rho^+$  and  $|N^-(v) \cap R| \in \rho^-$ , then  $\text{Tab}_\ell[\{\emptyset\}, \mathcal{Q}] = 0$ .

Before we proceed with the description of the algorithm updating the table entries at internal nodes, we give one more auxiliary observation.

*Observation 5.11.* Let  $d \in \mathbb{N}$ , and let  $G$  be a digraph with a 3-partition  $(A, B, W)$  of  $V(G)$ . For each equivalence class  $\mathcal{Q}_a$  of  $\equiv_{d,A}^\pm$  and each equivalence class  $\mathcal{Q}_b$  of  $\equiv_{d,B}^\pm$ , the following holds. There is an equivalence class  $\mathcal{Q}$  of  $\equiv_{d,A \cup B}^\pm$  such that for all  $R_a \in \mathcal{Q}_a$  and  $R_b \in \mathcal{Q}_b$ ,  $R_a \cup R_b \in \mathcal{Q}$ . Moreover, given a description of  $\mathcal{Q}_a$  and a description of  $\mathcal{Q}_b$ , we can compute a description of  $\mathcal{Q}$  in time  $\mathcal{O}(|W|)$ .

*Proof.* The first statement follows immediately from the definitions. For the second statement, let  $D = \text{desc}(\mathcal{Q})$ ,  $D_a = \text{desc}(\mathcal{Q}_a)$ , and  $D_b = \text{desc}(\mathcal{Q}_b)$ . Then, for all  $v \in W$ , we have  $D(v) = D_a(v) + D_b(v)$ .  $\square$

**Internal nodes of  $T$ .** Let  $t \in V(T)$  be an internal node with children  $a$  and  $b$ .

1. Consider each triple  $\mathcal{Q}_a, \mathcal{Q}_b, \mathcal{Q}_{\bar{t}}$  of equivalence classes of  $\equiv_{d,V_a}^\pm$ ,  $\equiv_{d,V_b}^\pm$ , and  $\equiv_{d,V_t}^\pm$ , respectively.
2. Let  $R_a \in \mathcal{Q}_a$ ,  $R_b \in \mathcal{Q}_b$ , and  $R_{\bar{t}} \in \mathcal{Q}_{\bar{t}}$ . Determine:
  - $\mathcal{Q}_{\bar{a}}$ , the equivalence class of  $\equiv_{d,V_a}^\pm$  containing  $R_b \cup R_{\bar{t}}$ .
  - $\mathcal{Q}_{\bar{b}}$ , the equivalence class of  $\equiv_{d,V_b}^\pm$  containing  $R_a \cup R_{\bar{t}}$ .
  - $\mathcal{Q}_t$ , the equivalence class of  $\equiv_{d,V_t}^\pm$  containing  $R_a \cup R_b$ .
3. Update  $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] = \text{opt}\{\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}], \text{Tab}_a[\mathcal{Q}_a, \mathcal{Q}_{\bar{a}}] + \text{Tab}_b[\mathcal{Q}_b, \mathcal{Q}_{\bar{b}}]\}$ .

The next two lemmas establish the correctness of the above algorithm.

**Lemma 5.12.** Let  $\Sigma, R$  be as above. Let  $G$  be a digraph and let  $(A, B, W)$  be a 3-partition of  $V(G)$ . Let  $S_a \subseteq A$ ,  $S_b \subseteq B$ , and  $S_w \subseteq W$ . Then  $(S_a, S_b \cup S_w)$   $(\Sigma, R)$ -dominates  $A$  and  $(S_b, S_a \cup S_w)$   $(\Sigma, R)$ -dominates  $B$  if and only if  $(S_a \cup S_b, S_w)$   $(\Sigma, R)$ -dominates  $A \cup B$ .

*Proof.* Suppose  $(S_a, S_b \cup S_w)$   $(\Sigma, R)$ -dominates  $A$  and  $(S_b, S_a \cup S_w)$   $(\Sigma, R)$ -dominates  $B$ ; let  $S = S_a \cup S_b \cup S_w$ . Then,  $S$   $\Sigma$ -dominates  $S_a$  and  $S_b$ , and therefore  $S$   $\Sigma$ -dominates  $S_a \cup S_b$ . Moreover,  $S$   $R$ -dominates  $A \setminus S_a$  and  $B \setminus S_b$ , so  $S$   $R$ -dominates  $(A \cup B) \setminus (S_a \cup S_b)$  which yields that  $(S_a \cup S_b, S_w)$   $(\Sigma, R)$ -dominates  $A \cup B$ . The other direction follows similarly.  $\square$

**Lemma 5.13.** For each node  $t \in V(T)$ , the table entries in  $\text{Tab}_t$  are computed correctly.

*Proof.* We prove the lemma by induction on the height of  $t$ . In the base case,  $t$  is a leaf. Correctness in this case is immediate. Suppose  $t$  is an internal node with children  $a$  and  $b$ .

For the first direction, assume that  $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] = k$  for an equivalence class  $\mathcal{Q}_t$  of  $\equiv_{d,V_t}^\pm$  and an equivalence class  $\mathcal{Q}_{\bar{t}}$  of  $\equiv_{d,V_t}^\pm$ . We show that in this case, there is a set  $S \in \mathcal{Q}_t$  of size  $k$  such that for all  $S_{\bar{t}} \in \mathcal{Q}_{\bar{t}}$ ,  $(S, S_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_t$ . By the update of the internal nodes and by the induction hypothesis, there are equivalence classes  $\mathcal{Q}_a$  of  $\equiv_{d,V_a}^\pm$ ,  $\mathcal{Q}_{\bar{a}}$  of  $\equiv_{d,V_a}^\pm$ ,  $\mathcal{Q}_b$  of  $\equiv_{d,V_b}^\pm$ , and  $\mathcal{Q}_{\bar{b}}$  of  $\equiv_{d,V_b}^\pm$  such that: There exist  $S_a \in \mathcal{Q}_a$ ,  $S_b \in \mathcal{Q}_b$  with  $|S_a| + |S_b| = k$  such that for all  $S_{\bar{a}} \in \mathcal{Q}_{\bar{a}}$  and  $S_{\bar{b}} \in \mathcal{Q}_{\bar{b}}$ ,  $(S_a, S_{\bar{a}})$   $(\Sigma, R)$ -dominates  $V_a$ , and  $(S_b, S_{\bar{b}})$   $(\Sigma, R)$ -dominates  $V_b$ . Additionally,  $S_a \cup R_{\bar{t}} \in \mathcal{Q}_{\bar{b}}$  and  $S_b \cup R_{\bar{t}} \in \mathcal{Q}_{\bar{a}}$ , where  $R_{\bar{t}} \in \mathcal{Q}_{\bar{t}}$ . Using Lemma 5.8, we conclude that  $(S_a, S_b \cup R_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_a$  and that  $(S_b, S_a \cup R_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_b$ . Lemma 5.12 yields that  $(S_a \cup S_b, R_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_a \cup V_b = V_t$ .

For the other direction, suppose that  $\text{opt} = \min$  and note that the case of  $\text{opt} = \max$  is analogous. We have to show for every pair of an equivalence class  $\mathcal{Q}_t$  of  $\equiv_{d, V_t}^\pm$  and an equivalence class  $\mathcal{Q}_{\bar{t}}$  of  $\equiv_{d, \bar{V}_t}^\pm$ , and  $R_{\bar{t}} \in \mathcal{Q}_{\bar{t}}$  that if there exists some  $S_t \in \mathcal{Q}_t$  of size at most  $k$  such that  $(S_t, R_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_t$ , then  $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] \leq |S_t|$ . Let  $S_a := S_t \cap V_a$  and  $S_b := S_t \cap V_b$ . At some point, the algorithm considered the equivalence classes  $\mathcal{Q}_a$  of  $\equiv_{d, V_a}^\pm$  and  $\mathcal{Q}_b$  of  $\equiv_{d, V_b}^\pm$  such that  $S_a \in \mathcal{Q}_a$  and  $S_b \in \mathcal{Q}_b$ . Since  $(S_a \cup S_b, R_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_a \cup V_b$ , it follows from Lemma 5.12 that  $(S_a, S_b \cup R_{\bar{t}})$   $(\Sigma, R)$ -dominates  $V_a$ . Note that by the above algorithm,  $S_b \cup R_{\bar{t}} \in \mathcal{Q}_{\bar{a}}$ , so by the induction hypothesis we have that  $\text{Tab}_a[\mathcal{Q}_a, \mathcal{Q}_{\bar{a}}] \leq |S_a|$ . Similarly we can deduce that  $\text{Tab}_b[\mathcal{Q}_b, \mathcal{Q}_{\bar{b}}] \leq |S_b|$ . Clearly,  $S_a \cup S_b = S_t \in \mathcal{Q}_t$ , so the algorithm above guarantees that  $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] \leq |S_t|$ .  $\square$

The algorithm described above results in the following theorem which is the first main result of this section.

**Theorem 5.14.** *Let  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$  be finite or co-finite,  $\Sigma = (\sigma^+, \sigma^-)$ ,  $R = (\rho^+, \rho^-)$ , and  $d = d(\Sigma, R)$ . There is an algorithm that given a digraph  $G$  on  $n$  vertices together with one of its branch decompositions  $(T, \mathcal{L})$ , computes and optimum-size  $(\Sigma, R)$ -dominating set in time  $\mathcal{O}(\text{nec}_d(T, \mathcal{L})^3 \cdot n^3 \log n)$ . For  $n \leq \text{nec}_d(T, \mathcal{L})$ , the algorithm runs in time  $\mathcal{O}(\text{nec}_d(T, \mathcal{L})^3 \cdot n^2)$ .*

*Proof.* We assume  $T$  to be rooted in an arbitrary node  $r \in V(T)$  of degree two, and do bottom-up dynamic programming along  $T$ . We first initialize the table entries at all nodes as described in Equation (2). At each node  $t \in V(T)$ , we perform the update of all table entries as described above in the corresponding paragraph, depending on whether  $t$  is a leaf or an internal node. We find the solution to the instance at hand at the table entry  $\text{Tab}_r[2^{V(G)}, \{\emptyset\}]$ .

Correctness of the algorithm follows from Lemma 5.13; we now analyze its runtime. We may assume that  $|V(T)| = \mathcal{O}(n)$ . By Lemma 5.4, we can compute all descriptions of the equivalence classes of all equivalence relations associated with the nodes of  $T$  in time at most  $\mathcal{O}(\text{nec}_d(T, \mathcal{L}) \log \text{nec}_d(T, \mathcal{L}) \cdot n^3) = \mathcal{O}(\text{nec}_d(T, \mathcal{L}) \cdot d \cdot n^3 \log n)$ . (Note that  $\log \text{nec}_d(T, \mathcal{L}) \leq d \log n$ .)

Initialization of all table entries takes time at most  $\mathcal{O}(\text{nec}_d(T, \mathcal{L})^2 \cdot n)$ . Updating the entries at leaf nodes takes time  $\mathcal{O}(d)$  per node, so at most  $\mathcal{O}(d \cdot n)$  in total, for all leaf nodes. At each internal node, we consider triples of equivalence classes, of which there are at most  $\text{nec}_d(T, \mathcal{L})^3$ . Once such a triple is fixed, the remaining computations can be done in time  $\mathcal{O}(n)$  by Observation 5.11, with an overhead of  $\mathcal{O}(d \log n)$  for querying the table entries, using a binary search tree. Therefore, for each triple of equivalence classes, the update takes at most  $\mathcal{O}(n + d \log n) = \mathcal{O}(dn)$  time; which implies that updating the table entries at all internal nodes takes time at most  $\mathcal{O}(\text{nec}_d(T, \mathcal{L})^3 \cdot d \cdot n^2)$ . Since  $d$  is a constant, we can bound the runtime of the algorithm by

$$\max\{\mathcal{O}(\text{nec}_d(T, \mathcal{L}) \cdot n^3 \log n), \mathcal{O}(\text{nec}_d(T, \mathcal{L})^3 \cdot n^2)\}. \quad \square$$

## Runtime in terms of bi-mim-width

We now show how to express the runtime of the algorithm from Theorem 5.14 as an XP-runtime parameterized by the bi-mim-width of  $(T, \mathcal{L})$ , in analogy with the case of undirected graphs [10]. The crucial observation is the following.

*Observation 5.15.* For  $d \in \mathbb{N}$ , a digraph  $G$ , and  $A \subseteq V(G)$ :  $\text{nec}(\equiv_{d, A}^\pm) \leq n^{d \cdot \text{bimim}_G(A)}$ .

*Proof.* By the same arguments given in [10] for the undirected case, we can show that

$$\text{nec}(\equiv_{d, A}^+) \leq n^{d \cdot \text{mim}_G^+(A)} \text{ and that } \text{nec}(\equiv_{d, A}^-) \leq n^{d \cdot \text{mim}_G^-(A)}.$$

Therefore,

$$\text{nec}(\equiv_{d,A}^\pm) = \text{nec}(\equiv_{d,A}^+) \cdot \text{nec}(\equiv_{d,A}^-) \leq n^{d \cdot \text{mim}_G^+(A)} \cdot n^{d \cdot \text{mim}_G^-(A)} = n^{d \cdot \text{bimim}_G(A)}. \quad \square$$

*Corollary 5.16.* Let  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$  be finite or co-finite,  $\Sigma = (\sigma^+, \sigma^-)$ ,  $R = (\rho^+, \rho^-)$ , and  $d = d(\Sigma, R)$ . Let  $G$  be a digraph on  $n$  vertices with branch decomposition  $(T, \mathcal{L})$  of bi-mim-width  $w \geq 1$ . There is an algorithm that given any such  $G$  and  $(T, \mathcal{L})$  computes an optimum-size  $(\Sigma, R)$ -dominating set in time  $\mathcal{O}(n^{3dw+2})$ .

### 5.3 Directed Vertex Partitioning Problems

We now show that the locally checkable vertex partitioning problems can be solved in XP time parameterized by the bi-mim-width of a given branch decomposition. In analogy with [14], we lift the  $d$ -bi-neighborhood equivalence to  $q$ -tuples over vertex sets, which allows for devising the desired dynamic programming algorithm. We omit several technical details as they are very similar to the ones in the previous section. We begin by recalling the definition of a bi-neighborhood constraint matrix.

**Definition 5.17.** A *bi-neighborhood-constraint* matrix is a  $(q \times q)$ -matrix  $D_q$  over pairs of finite or co-finite sets of natural numbers. Let  $G$  be a digraph, and  $\mathcal{X} = (X_1, \dots, X_q)$  be a  $q$ -partition of  $V(G)$ . We say that  $\mathcal{X}$  is a *D-partition* if for all  $i, j \in \{1, \dots, q\}$  with  $D_q[i, j] = (\mu_{i,j}^+, \mu_{i,j}^-)$ , we have that for all  $v \in X_i$ ,  $|N^+(v) \cap X_j| \in \mu_{i,j}^+$  and  $|N^-(v) \cap X_j| \in \mu_{i,j}^-$ . The *d-value* of  $D_q$  is  $d(D_q) = \max_{i,j} \{d(\mu_{i,j}^+), d(\mu_{i,j}^-)\}$ .

**Definition 5.18.** Let  $G$  be a digraph and  $A \subseteq V(G)$ . Two  $q$ -tuples of subsets of  $A$ ,  $\mathcal{X} = (X_1, \dots, X_q)$  and  $\mathcal{Y} = (Y_1, \dots, Y_q)$ , are *d-bi-neighborhood equivalent* w.r.t.  $A$ , if

$$\begin{aligned} \forall i \forall v \in \overline{A}: \min\{d, |N^+(v) \cap X_i|\} &= \min\{d, |N^+(v) \cap Y_i|\} \text{ and} \\ \min\{d, |N^-(v) \cap X_i|\} &= \min\{d, |N^-(v) \cap Y_i|\}. \end{aligned}$$

In this case we write  $\mathcal{X} \equiv_{q,d,A}^\pm \mathcal{Y}$ .

*Observation 5.19.* Let  $G$  be a digraph,  $A \subseteq V(G)$ , and let  $\mathcal{X} = (X_1, \dots, X_q) \in (2^A)^q$  and  $\mathcal{Y} = (Y_1, \dots, Y_q) \in (2^A)^q$ . Then,  $\mathcal{X} \equiv_{q,d,A}^\pm \mathcal{Y}$  if and only if for all  $i \in \{1, \dots, q\}$ ,  $X_i \equiv_{d,A}^\pm Y_i$ . Therefore,  $\text{nec}(\equiv_{q,d,A}^\pm) = \text{nec}(\equiv_{d,A}^\pm)^q$ .

For a  $q$ -tuple  $\mathcal{X} = (X_1, \dots, X_q)$  of subsets of  $A$ , the  $(q, d)$ -bi-neighborhood of  $\mathcal{X}$  w.r.t.  $A$  is  $U_{q,d,A}^\pm(\mathcal{X}) = (U_{d,A}^\pm(X_1), \dots, U_{d,A}^\pm(X_q))$ , and we denote the set of all  $(q, d)$ -bi-neighborhoods w.r.t.  $A$  by  $\mathcal{U}_{q,d,A}^\pm$ . By the previous observation, for any  $A \subseteq V(G)$ , there is a natural bijection between the  $(q, d)$ -bi-neighborhoods w.r.t.  $A$  and the equivalence classes of  $\equiv_{q,d,A}^\pm$ . Furthermore, we can enumerate  $\mathcal{U}_{q,d,A}^\pm$  by invoking the algorithm of Lemma 5.4  $q$  times and then generating all  $q$ -tuples of  $\mathcal{U}_{d,A}^\pm$ .

*Corollary 5.20.* Let  $G$  be a digraph on  $n$  vertices,  $A \subseteq V(G)$ , and  $q, d \in \mathbb{N}$  with  $q \geq 2$ . There is an algorithm that enumerates all members of  $\mathcal{U}_{q,d,A}^\pm$  in time  $\mathcal{O}(\text{nec}(\equiv_{d,A}^\pm)^q \cdot qdn^2)$ . Furthermore, for each  $\mathcal{Y} \in \mathcal{U}_{q,d,A}^\pm$ , the algorithm provides some  $\mathcal{X} \in (2^A)^q$  with  $U_{q,d,A}^\pm(\mathcal{X}) = \mathcal{Y}$ .

**Definition 5.21.** Let  $D_q$  be a bi-neighborhood constraint matrix. Let  $G$  be a digraph and  $A \subseteq V(G)$ . Let  $\mathcal{X} = (X_1, \dots, X_q) \in (2^A)^q$  and  $\mathcal{Y} = (Y_1, \dots, Y_q) \in (2^A)^q$ . We say that  $(\mathcal{X}, \mathcal{Y})$   $D_q$ -dominates  $A$  if for all  $i, j \in \{1, \dots, q\}$ ,  $X_i \cup Y_j$   $D_q[i, j]$ -dominates  $A$ .

For an equivalence class  $\mathcal{Q}$  of  $\equiv_{q,d,A}^\pm$ , its *description*,  $\text{desc}(\mathcal{Q})$ , is the  $(q, d)$ -bi-neighborhood of all its members. Again we index the table entries with descriptions of equivalence classes. For a clearer presentation we will also here skip explicit mentions of the  $\text{desc}$ -operator.

**Definition 5.22.** Let  $D_q$  be a bi-neighborhood constraint matrix with  $d(D_q) = d$ , and let  $G$  be a digraph with branch decomposition  $(T, \mathcal{L})$ , and  $t \in V(T)$ . Let  $\mathcal{Q}_t$  be an equivalence class of  $\equiv_{q,d,V_t}^\pm$  and  $\mathcal{Q}_{\bar{t}}$  be an equivalence class of  $\equiv_{q,d,\bar{V}_t}^\pm$ . Then,

$$\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] := \begin{cases} \text{True}, & \text{if } \exists q\text{-partition } \mathcal{S} \text{ of } V_t \text{ such that:} \\ & \mathcal{S} \in \mathcal{Q}_t \text{ and for all } \mathcal{R}_{\bar{t}} \in \mathcal{Q}_{\bar{t}}: (\mathcal{S}, \mathcal{R}_{\bar{t}}) \text{ } D_q\text{-dominates } V_t; \\ \text{False}, & \text{otherwise.} \end{cases}$$

By the definition of the table entries we have that  $G$  has a  $D_q$ -partition if and only if some entry in  $\text{Tab}_r$  is true, where  $r$  is the root of the given branch decomposition of  $G$ . We now describe the algorithm. Initially, we set all table entries at all nodes to **False**.

**Leaves of  $T$ .** If  $\ell \in V(T)$  is a leaf of  $T$ , then let  $v \in V(G)$  be such that  $\mathcal{L}(v) = \ell$ . We have to consider the following  $q$   $q$ -partitions of  $\{v\}$  (recall that parts of a partition may be empty): For  $i \in \{1, \dots, q\}$ , we have to consider the partition  $\mathcal{X}_i = (X_1, \dots, X_q)$  where  $X_i = \{v\}$  and for  $j \neq i$ ,  $X_j = \emptyset$ . While these partitions are equal up to renaming, they might differ with respect to  $D_q$ . We have that  $U_{q,d,\{v\}}^\pm(\mathcal{X}_i)$  is the all-zeroes vector in coordinates  $j \neq i$ , and  $U_{d,A}^\pm(\{v\})$  in coordinate  $i$ . We denote the corresponding equivalence class of  $\equiv_{q,d,\{v\}}^\pm$  by  $\mathcal{Q}_i$ . For each equivalence class  $\mathcal{Q}_{\bar{\ell}}$  of  $\equiv_{q,d,V(G) \setminus \{v\}}^\pm$ , let  $\mathcal{Y} = (Y_1, \dots, Y_\ell)$  be one of its elements. We then perform the following updates: For all  $j$ , let  $(\mu_{i,j}^+, \mu_{i,j}^-) = D_q[i, j]$ ; then

$$\text{Tab}_\ell[\mathcal{Q}_i, \mathcal{Q}_{\bar{\ell}}] = \text{True} \text{ if } \forall j: |N^+(v) \cap Y_j| \in \mu_{i,j}^+ \text{ and } |N^-(v) \cap Y_j| \in \mu_{i,j}^-.$$

In the following, for two  $q$ -tuples  $(X_1, \dots, X_q)$  and  $(Y_1, \dots, Y_q)$ , we denote their coordinate-wise union as  $(X_1, \dots, X_q) \bigcup_q (Y_1, \dots, Y_q) = (X_1 \cup Y_1, \dots, X_q \cup Y_q)$ .

**Internal nodes of  $T$ .** Let  $t \in V(T)$  be an internal node with children  $a$  and  $b$ .

1. Consider each triple  $\mathcal{Q}_a, \mathcal{Q}_b, \mathcal{Q}_{\bar{t}}$  of equivalence classes of  $\equiv_{q,d,V_a}^\pm, \equiv_{q,d,V_b}^\pm$ , and  $\equiv_{q,d,\bar{V}_t}^\pm$ , respectively.
2. Let  $R_a \in \mathcal{Q}_a$ ,  $R_b \in \mathcal{Q}_b$ , and  $R_{\bar{t}} \in \mathcal{Q}_{\bar{t}}$ . Determine:
  - $\mathcal{Q}_{\bar{a}}$ , the equivalence class of  $\equiv_{q,d,\bar{V}_a}^\pm$  containing  $R_b \bigcup_q R_{\bar{t}}$ .
  - $\mathcal{Q}_{\bar{b}}$ , the equivalence class of  $\equiv_{q,d,\bar{V}_b}^\pm$  containing  $R_a \bigcup_q R_{\bar{t}}$ .
  - $\mathcal{Q}_t$ , the equivalence class of  $\equiv_{q,d,V_t}^\pm$  containing  $R_a \bigcup_q R_b$ .
3. If  $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] = \text{False}$ , then update  $\text{Tab}_t[\mathcal{Q}_t, \mathcal{Q}_{\bar{t}}] = \text{Tab}_a[\mathcal{Q}_a, \mathcal{Q}_{\bar{a}}] \wedge \text{Tab}_b[\mathcal{Q}_b, \mathcal{Q}_{\bar{b}}]$ .



Applying the arguments given in the proof of Lemma 5.13 to each part of the corresponding partitions yields the correctness of the resulting algorithm. Also the running time can be analyzed in a similar way, using Corollary 5.20 and Observation 5.19 to bound the complexity of enumerating all equivalence classes of the equivalence relations  $\equiv_{q,d,\cdot}^\pm$ . We have the following theorem.

**Theorem 5.23.** *Let  $D_q$  be a bi-neighborhood constraint matrix with  $d = d(D_q)$ . There is an algorithm that given a digraph  $G$  on  $n$  vertices together with one of its branch decompositions  $(T, \mathcal{L})$ , determines whether  $G$  has a  $D_q$ -partition in time  $\mathcal{O}(\text{nec}_d(T, \mathcal{L})^{3q} \cdot q \cdot n^3 \log n)$ . For  $n \leq \text{nec}_d(T, \mathcal{L})$ , the algorithm runs in time  $\mathcal{O}(\text{nec}_d(T, \mathcal{L})^{3q} \cdot q \cdot n^2)$ .*

Combining the previous theorem with Observation 5.15 gives the following algorithms parameterized by the bi-mim-width of a given branch decomposition.

*Corollary 5.24.* *Let  $D_q$  be a bi-neighborhood constraint matrix with  $d = d(D_q)$ . Let  $G$  be a digraph on  $n$  vertices with branch decomposition  $(T, \mathcal{L})$  of bi-mim-width  $w$ . There is an algorithm that given any such  $G$  and  $(T, \mathcal{L})$  decides whether  $G$  has a  $D_q$ -partition in time  $\mathcal{O}(q \cdot n^{3qdw+2})$ .*

## 5.4 Distance- $r$ Variants

We now turn to distance variants of all problems considered in this section so far. For instance, for  $r \in \mathbb{N}$ , the DISTANCE- $r$  DOMINATING SET problem asks for a minimum size set  $S$  of vertices of a digraph  $G$ , such that each vertex in  $V(G) \setminus S$  is at distance at most  $r$  from a vertex in  $S$ . Note that for  $r = 1$ , we recover the DOMINATING SET problem. We can generalize all DLCV problems to their distance-versions.

**Definition 5.25.** Let  $G$  be a digraph. For  $r \in \mathbb{N}$ , the  $r$ -out-ball of a vertex  $v \in V(G)$  is the set of vertices  $B_r^+(v) = \{w \in V(G) \mid \text{dist}_G(v, w) \leq r\}$ , and the  $r$ -in-ball of a vertex  $v$  is the set of vertices  $B_r^-(v) = \{u \in V(G) \mid \text{dist}_G(u, v) \leq r\}$ .

In distance- $r$  versions of DLCV problems, restrictions are posed on  $B_r^+(v)$  instead of  $N^+(v)$  and on  $B_r^-(v)$  instead of  $N^-(v)$ .

**Definition 5.26.** Let  $r \in \mathbb{N}$ ; let  $\sigma^+, \sigma^-, \rho^+, \rho^-$  be finite or co-finite subsets of  $\mathbb{N}$ , let  $\Sigma = (\sigma^+, \sigma^-)$  and  $R = (\rho^+, \rho^-)$ . Let  $G$  be a digraph and  $S \subseteq V(G)$ . We say that  $S$  *distance- $r$   $(\Sigma, R)$ -dominates*  $G$ , if:

$$\forall v \in V(G): |B_r^+(v) \cap S| \in \begin{cases} \sigma^+, & \text{if } v \in S \\ \rho^+, & \text{if } v \notin S \end{cases} \quad \text{and} \quad |B_r^-(v) \cap S| \in \begin{cases} \sigma^-, & \text{if } v \in S \\ \rho^-, & \text{if } v \notin S \end{cases}$$

It is not difficult to see that a set  $S \subseteq V(G)$  is a distance- $r$   $(\Sigma, R)$ -dominating set in  $G$  if and only if  $S$  is a  $(\Sigma, R)$ -dominating set in  $G^r$ , the  $r$ -th power of  $G$ . Therefore, to solve DISTANCE- $r$   $(\Sigma, R)$ -SET on  $G$ , we can simply compute  $G^r$  and solve  $(\Sigma, R)$ -SET on  $G^r$ . By Lemma 3.9 and Corollary 5.16, we have the following consequence.

*Corollary 5.27.* *Let  $r \in \mathbb{N}$ ; let  $\sigma^+, \sigma^-, \rho^+, \rho^- \subseteq \mathbb{N}$  be finite or co-finite,  $\Sigma = (\sigma^+, \sigma^-)$ ,  $R = (\rho^+, \rho^-)$ , and  $d = d(\Sigma, R)$ . Let  $G$  be a digraph on  $n$  vertices with branch decomposition  $(T, \mathcal{L})$  of bi-mim-width  $w \geq 1$ . There is an algorithm that given any such  $G$  and  $(T, \mathcal{L})$  computes an optimum-size distance- $r$   $(\Sigma, R)$ -dominating set in time  $\mathcal{O}(n^{3drw+2})$ .*

**Definition 5.28.** Let  $D$  be a  $(q \times q)$  bi-neighborhood-constraint matrix. Let  $G$  be a digraph. A  $q$ -partition  $\mathcal{X} = (X_1, \dots, X_q)$  of  $V(G)$  is a *distance- $r$   $D$ -partition* of  $G$ , if for all  $i, j$ , where  $D[i, j] = (\mu_{i,j}^+, \mu_{i,j}^-)$ , we have that for all  $v \in X_i$ ,  $|B_r^+(v) \cap X_j| \in \mu_{i,j}^+$  and  $|B_r^-(v) \cap X_j| \in \mu_{i,j}^-$ .

By similar reasoning as above and Lemma 3.9 and Corollary 5.24, we have the following.

*Corollary 5.29.* *Let  $r \in \mathbb{N}$ ; let  $D_q$  be a bi-neighborhood constraint matrix with  $d = d(D_q)$ . Let  $G$  be a digraph on  $n$  vertices with branch decomposition  $(T, \mathcal{L})$  of bi-mim-width  $w$ . There is an algorithm that given any such  $G$  and  $(T, \mathcal{L})$  decides whether  $G$  has a distance- $r$   $D_q$ -partition in time  $\mathcal{O}(q \cdot n^{3qdrw+2})$ .*

## 6 Conclusion

We introduced the digraph width measure bi-mim-width, and showed that directed locally checkable vertex problems (and their distance- $r$  versions) can be solved in polynomial time if the input digraph is given together with a branch decomposition of constant bi-mim-width. A natural next step in the understanding of this new parameter would be to determine the complexity of the DIRECTED FEEDBACK VERTEX SET problem on digraphs of bounded bi-mim-width. We showed that several classes of intersection digraphs have constant bi-mim-width which adds a large number of polynomial-time algorithms for locally checkable problems related to domination and independence (given a representation) to the relatively sparse literature on the subject.

Intersection digraph classes such as interval digraphs seem too complex to give polynomial-time algorithms for optimization problems. Our work points to reflexivity as a reasonable additional restriction to give successful algorithmic applications of intersection digraphs, while maintaining a high degree of generality. This was observed independently for interval digraphs by Francis, Hell, and Jacob [22] who studied the KERNEL, ABSORBING SET, and DOMINATING SET problems. Apart from giving polynomial-time algorithms for these problems on reflexive interval digraphs, they showed that even for the severely restricted case when the intervals associated with the vertices are single points, the aforementioned problems remain hard. Towards a better understanding of the role of reflexivity in obtaining algorithms for intersection digraphs, we ask if reflexive permutation digraphs and reflexive rooted directed path digraphs have bounded bi-mim-width, as we only proved bounds on the bi-mim-width of subclasses of these digraph classes.

As our algorithms rely on a representation of the input digraphs being provided at the input, we are naturally interested in computing representations of intersection digraph classes of bounded bi-mim-width in polynomial time. So far, this is only known for (reflexive) interval digraphs.

In Lemma 3.9, we proved that the  $r$ -th power of a digraph of bi-mim-width  $w$  has bi-mim-width at most  $rw$ . For undirected graphs, there is a bound that does not depend on  $r$  [28]. We leave an open question whether a bound that does not depend on  $r$  exists.

## References

- [1] Noga Alon, Jørgen Bang-Jensen, and Stéphane Bessy. Out-colourings of digraphs. *J. Graph Theory*, 93(1):88–112, 2020.
- [2] S. Arumugam, K. Jacob, and Lutz Volkmann. Total and connected domination in digraphs. *Australas. J Comb.*, 39:283–292, 2007.
- [3] Jørgen Bang-Jensen, Stéphane Bessy, Frédéric Havet, and Anders Yeo. Out-degree reducing partitions of digraphs. *Theor. Comput. Sci.*, 719:64–72, 2018.
- [4] Jørgen Bang-Jensen, Stéphane Bessy, Frédéric Havet, and Anders Yeo. Bipartite spanning sub(di)graphs induced by 2-partitions. *J. Graph Theory*, 92(2):130–151, 2019.

- [5] Jørgen Bang-Jensen and Tilde My Christiansen. Degree constrained 2-partitions of semicomplete digraphs. *Theor. Comput. Sci.*, 746:112–123, 2018.
- [6] Jørgen Bang-Jensen, Nathann Cohen, and Frédéric Havet. Finding good 2-partitions of digraphs II. Enumerable properties. *Theor. Comput. Sci.*, 640:1–19, 2016.
- [7] Jørgen Bang-Jensen and Gregory Gutin, editors. *Classes of directed graphs*. Springer Monographs in Mathematics. Springer, Cham, 2018.
- [8] David W. Bange, Anthony E. Barkauskas, Linda H. Host, and Lane H. Clark. Efficient domination of the orientations of a graph. *Discret. Math.*, 178(1-3):1–14, 1998.
- [9] Lowell W. Beineke and Christina M. Zamfirescu. Connection digraphs and second-order line digraphs. *Discrete Math.*, 39(3):237–254, 1982.
- [10] Rémy Belmonte and Martin Vatshelle. Graph classes with structured neighborhoods and algorithmic applications. *Theor. Comput. Sci.*, 511:54–65, 2013.
- [11] Miklós Biró, Mihály Hujter, and Zsolt Tuza. Precoloring extension. I. Interval graphs. *Discret. Math.*, 100(1-3):267–279, 1992.
- [12] Flavia Bonomo-Braberman, Nick Brettell, Andrea Munaro, and Daniël Paulusma. Solving problems on generalized convex graphs via mim-width. In Anna Lubiw and Mohammad R. Salavatipour, editors, *Proceedings of the 17th International Symposium on Algorithms and Data Structures (WADS 2021)*, volume 12808 of *Lecture Notes in Computer Science*, pages 200–214. Springer, 2021.
- [13] Andreas Brandstädt, Van Bang Le, and Jeremy P. Spinrad. *Graph classes: a survey*. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.
- [14] 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.
- [15] Domingos Moreira Cardoso, Marcin Kaminski, and Vadim V. Lozin. Maximum  $k$ -regular induced subgraphs. *J. Comb. Optim.*, 14(4):455–463, 2007.
- [16] Michael Cary, Jonathan Cary, and Savari Prabhu. Independent domination in directed graphs. *Communications in Combinatorics and Optimization*, 6(1):67–80, 2021.
- [17] Gary Chartrand, Peter Dankelmann, Michelle Schultz, and Henda C. Swart. Twin domination in digraphs. *Ars Comb.*, 67, 2003.
- [18] Bruno Courcelle. The monadic second order logic of graphs VI: on several representations of graphs by relational structures. *Discret. Appl. Math.*, 54(2-3):117–149, 1994.
- [19] Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, Heidelberg, fourth edition, 2010.
- [20] Tomás Feder, Pavol Hell, Jing Huang, and Arash Rafiey. Interval graphs, adjusted interval digraphs, and reflexive list homomorphisms. *Discrete Appl. Math.*, 160(6):697–707, 2012.

- [21] Fedor V. Fomin, Petr A. Golovach, and Jean-Florent Raymond. On the tractability of optimization problems on  $H$ -graphs. *Algorithmica*, 82(9):2432–2473, 2020.
- [22] Mathew C. Francis, Pavol Hell, and Dalu Jacob. On the kernel and related problems in interval digraphs. *CoRR*, abs/2107.08278, 2021. To appear at ISAAC 2021.
- [23] Yumin Fu. Dominating set and converse dominating set of a directed graph. *The American Mathematical Monthly*, 75(8):861–863, 1968.
- [24] Fănică Gavril. A recognition algorithm for the intersection graphs of directed paths in directed trees. *Discrete Math.*, 13(3):237–249, 1975.
- [25] J. Ghoshal, Renu Laskar, and D. Pillone. Topics on domination in directed graphs. In Theresa W. Haynes, Stephen T. Hedetniemi, and Peter J. Slater, editors, *Domination in Graphs: Advanced Topics*. Taylor & Francis, 2017.
- [26] Pavol Hell and Jaroslav Nešetřil. *Graphs and homomorphisms*, volume 28 of *Oxford lecture series in mathematics and its applications*. Oxford University Press, 2004.
- [27] Svein Høgemo, Jan Arne Telle, and Erlend Raa Vågset. Linear mim-width of trees. In Ignasi Sau and Dimitrios M. Thilikos, editors, *Graph-Theoretic Concepts in Computer Science*, pages 218–231, Cham, 2019. Springer International Publishing.
- [28] Lars Jaffke, O-joung Kwon, Torstein J. F. Strømme, and Jan Arne Telle. Mim-width III. Graph powers and generalized distance domination problems. *Theoret. Comput. Sci.*, 796:216–236, 2019.
- [29] Thor Johnson, Neil Robertson, P. D. Seymour, and Robin Thomas. Directed tree-width. *J. Combin. Theory Ser. B*, 82(1):138–154, 2001.
- [30] Dong Yeap Kang, O-joung Kwon, Torstein J. F. Strømme, and Jan Arne Telle. A width parameter useful for chordal and co-comparability graphs. *Theoret. Comput. Sci.*, 704:1–17, 2017.
- [31] Mamadou Moustapha Kanté. The rank-width of directed graphs. *preprint*, arxiv.org/abs/0709.1433, 2007.
- [32] Mamadou Moustapha Kanté and Michael Rao. The rank-width of edge-coloured graphs. *Theory Comput. Syst.*, 52(4):599–644, 2013.
- [33] Stefan Mengel. Lower bounds on the mim-width of some graph classes. *Discrete Appl. Math.*, 248:28–32, 2018.
- [34] Haiko Müller. Recognizing interval digraphs and interval bigraphs in polynomial time. *Discrete Appl. Math.*, 78(1-3):189–205, 1997.
- [35] Lyes Ouldrabah, Mostafa Blidia, and Ahmed Bouchou. On the  $k$ -domination number of digraphs. *J. Comb. Optim.*, 38(3):680–688, 2019.
- [36] Sang-il Oum. Rank-width and vertex-minors. *J. Comb. Theory, Ser. B*, 95(1):79–100, 2005.
- [37] Sang-il Oum. Rank-width is less than or equal to branch-width. *J. Graph Theory*, 57(3):239–244, 2008.

- [38] Erich Prisner. Algorithms for interval catch digraphs. *Discret. Appl. Math.*, 51(1-2):147–157, 1994.
- [39] Amina Ramoul and Mostafa Blidia. A new generalization of kernels in digraphs. *Discret. Appl. Math.*, 217:673–684, 2017.
- [40] Oliver Schaudt. Efficient total domination in digraphs. *J. Discrete Algorithms*, 15:32–42, 2012.
- [41] M. Sen, S. Das, A. B. Roy, and D. B. West. Interval digraphs: an analogue of interval graphs. *J. Graph Theory*, 13(2):189–202, 1989.
- [42] M. Sen, S. Das, and Douglas B. West. Circular-arc digraphs: a characterization. *J. Graph Theory*, 13(5):581–592, 1989.
- [43] Petra Smolíková. The simple chromatic number of oriented graphs. *Electronic Notes in Discrete Mathematics*, 5:281–283, 2000.
- [44] Eric Sopena. The chromatic number of oriented graphs. *J. Graph Theory*, 25(3):191–205, 1997.
- [45] Éric Sopena. Homomorphisms and colourings of oriented graphs: An updated survey. *Discret. Math.*, 339(7):1993–2005, 2016.
- [46] Jan Arne Telle and Andrzej Proskurowski. Algorithms for vertex partitioning problems on partial  $k$ -trees. *SIAM J. Discrete Math.*, 10(4):529–550, 1997.
- [47] Martin Vatshelle. *New Width Parameters of Graphs*. PhD thesis, Univ. Bergen, 2012.
- [48] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, New Jersey, 1944.