The Parameterized Complexity of Coloring Mixed Graphs

Antonio Lauerbach
Julius-Maximilians-Universität Würzburg, Germany

Konstanty Junosza-Szaniawski
Warsaw University of Technology, Poland

Marie Diana Sieper
Julius-Maximilians-Universität Würzburg, Germany

Alexander Wolff
Julius-Maximilians-Universität Würzburg, Germany

Abstract

A mixed graph contains (undirected) edges as well as (directed) arcs, thus generalizing undirected and directed graphs. A proper coloring c of a mixed graph G assigns a positive integer to each vertex such that $c(u) \neq c(v)$ for every edge $\{u, v\}$ and $c(u) < c(v)$ for every arc (u, v) of G. As in classical coloring, the objective is to minimize the number of colors. Thus, mixed (graph) coloring generalizes classical coloring of undirected graphs and allows for more general applications, such as scheduling with precedence constraints, modeling metabolic pathways, and process management in operating systems; see a survey by Sotskov [Mathematics, 2020].

We initiate the systematic study of the parameterized complexity of mixed coloring. We focus on structural graph parameters that lie between cliquewidth and vertex cover, primarily with respect to the underlying undirected graph. Unlike classical coloring, which is fixed-parameter tractable (FPT) parameterized by treewidth or neighborhood diversity, we show that mixed coloring is W[1]-hard for treewidth and even paraNP-hard for neighborhood diversity. To utilize the directedness of arcs, we introduce and analyze natural generalizations of neighborhood diversity and cliquewidth to mixed graphs, and show that mixed coloring becomes FPT when parameterized by (the generalized) mixed neighborhood diversity. Further, we investigate how these parameters are affected if we add transitive arcs, which do not affect colorings. Finally, we provide tight bounds on the chromatic number of mixed graphs, generalizing known bounds on mixed interval graphs.

2012 ACM Subject Classification Theory of computation → Parameterized complexity and exact algorithms; Mathematics of computing → Graph coloring

Keywords and phrases Mixed Graphs, Coloring, Parameterized Complexity, Structural Graph Parameters

Digital Object Identifier 10.4230/LIPIcs.SWAT.2026.28

Related Version *Full Version*: https://arxiv.org/abs/2604.15274 [26]

Acknowledgements This work was started at the workshop Homonolo 2024 in Nová Louka. We thank the organizers and the other participants.

1 Introduction

The problem of coloring graphs, i.e., assigning as few colors as possible to vertices of an undirected graph such that adjacent vertices receive distinct colors, is an old and extensively studied problem in graph theory with applications in various areas, such as scheduling, frequency assignment, and graph drawing [24]. An example of a scheduling problem that can be modeled using graph coloring is timetabling, where courses (vertices) need to be scheduled in certain time slots (colors) while respecting conflicts (edges), e.g., due to two courses having the same teacher or students. These types of problems are referred to as chromatic scheduling problems. In real world applications, there are usually several additional constraints, such as certain courses having to be scheduled in specific time slots, certain teachers not being available at certain times, courses using multiple consecutive time slots, or a lecture having to be held before the corresponding exercises. Thus, there exist several extensions of coloring to accommodate these additional constraints, such as using list coloring to restrict courses to certain slots [9], or using interval coloring to enforce consecutive time slots [23]. To account for precedence constraints, for example, if a course has to be scheduled before another, Hansen et al. [20] added arcs to the graph, with the idea that the arc (u, v) expresses that u has to be scheduled before v. This leads to the notion of a *mixed graph* G, which consists of a set $V(G)$ of *vertices*, a set $E(G)$ of (undirected) *edges*, and a set $A(G)$ of (directed) *arcs*. A mixed graph combines the concepts of undirected and directed graphs. We require mixed graphs to be *simple*, i.e., without loops and without parallel edges or arcs.

A *k-coloring* $c\colon V(G) \to \{1, \ldots, k\}$ of a mixed graph G is *proper* if it holds for every edge $\{u, v\} \in E(G)$ that $c(u) \neq c(v)$ and it holds for every arc $(u, v) \in A(G)$ that $c(u) < c(v)$. We call the problem of deciding whether a mixed graph can be properly colored with k colors k-MixedColoring. As a mixed graph cannot be properly colored if it contains a directed cycle, we consider only mixed graphs without directed cycles. Apart from scheduling, mixed graph colorings have also been applied in graph drawing in order to compact layered orthogonal drawings [19]. See the survey by Sotskov [32] for more details and applications of mixed graph colorings.

Since MixedColoring generalizes the classical Coloring problem for undirected graphs, it is also NP-hard for three or more colors. As such, it is natural to study its parameterized complexity. However, to our knowledge, there has not yet been a systematic study of the parameterized complexity of MixedColoring. The only known results in this direction are an FPT-algorithm parameterized by the number of edges (i.e., not counting arcs) by Hansen et al. [20], with subsequent improvements by Damaschke [8], as well as an XP-algorithm parameterized by treewidth by Ries and de Werra [30].

Meanwhile, the parameterized complexity of Coloring has been extensively studied. On the positive side, results include the well-known dynamic program that runs in $\mathcal{O}^*(k^{\mathsf{tw}})$ time, where k is the number of colors, tw is the treewidth of the graph, and the $\mathcal{O}^*$-notation hides polynomial factors. Since treewidth bounds the chromatic number of undirected graphs, this implies that Coloring is FPT parameterized by treewidth. Lokshtanov et al. [27] even showed that under SETH (a hypothesis about the complexity of SAT with at most k literals per clause) there is no $\mathcal{O}^*((k-\varepsilon)^{\mathsf{tw}})$-time algorithm for any $\varepsilon > 0$, i.e., the $\mathcal{O}^*(k^{\mathsf{tw}})$ algorithm is probably optimal. Further results include Coloring being FPT parameterized by neighborhood diversity, as shown by Ganian [15]. On the negative side, Fomin et al. [13] showed that Coloring is W[1]-hard parameterized by cliquewidth. However, Coloring still admits an XP-algorithm parameterized by cliquewidth, as shown by Kobler and Rotics [22]. Further, applying Courcelle's theorem [4, 5], a famous meta-theorem that yields FPT-algorithms, it follows that Coloring is FPT parameterized by cliquewidth plus chromatic number.

Understanding the relationships between various structural graph parameters is crucial for analyzing the parameterized complexity of a problem. We say that a graph parameter α *(upper) bounds* another parameter β if there exists a computable function f such that for every (mixed) graph G, it holds that $\beta(G) \leqslant f(\alpha(G))$. If no such function exists, we say that α *does not (upper) bound* β. If α (upper) bounds β and β does not (upper) bound α, we say that α *strictly (upper) bounds* β. Meanwhile, if neither parameter bounds the other, we

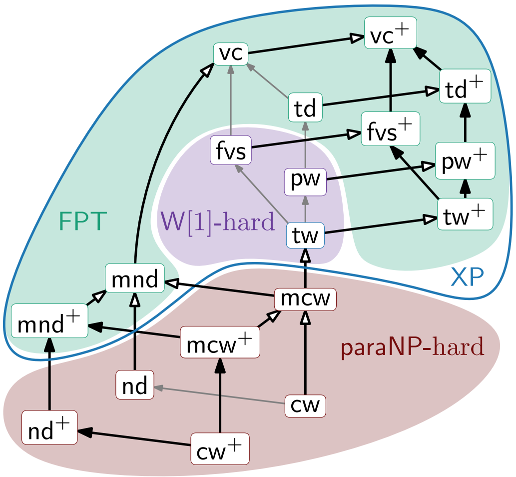

Figure 1 Overview of the hierarchy of structural parameters of mixed graphs (without directed cycles), and the parameterized complexity landscape for MixedColoring. Arrows represent strict (upper) bounds between parameters. Bold arrows indicate new relationships established in this work. White-tipped arrows highlight results that are not straightforward consequences of known results. Composite parameters are omitted for clarity.

Table 1 Our parameterized complexity results for MixedColoring including results for composite parameters. The complexity of other parameters can be inferred from the parameter hierarchy in Figure 1.

Parameter	Complexity	Reference
mcw + χ	FPT	Thm. 18
mnd	FPT	Thm. 28
nd + χ	FPT	Thm. 31
tw	XP	Cor. 20
fvs + pw	W[1]-hard	Thm. 23
mcw + Λ	paraNP-hard	Thm. 24
nd$^+$	paraNP-hard	Thm. 25

say that they are *incomparable*. Conversely, if two parameters bound each other, we say that they are *equivalent*. For example, since treewidth is bounded by vertex cover, this implies that any problem that is FPT parameterized by treewidth, such as Coloring, is also FPT parameterized by vertex cover.

For mixed graphs however, it is not immediately clear how to apply the existing parameters, as they are defined for undirected graphs. The approach taken in previous works is to consider the parameters w.r.t. the *underlying undirected graph*, i.e., the graph obtained by replacing all arcs with edges. The downside of this approach is that it neglects the directedness of arcs. In fact, Courcelle and Olariu [6] originally defined cliquewidth for undirected as well as for directed graphs, yielding a natural generalization for mixed graphs. Further, Fernau et al. [11] generalized neighborhood diversity to directed graphs by distinguishing between incoming and outgoing neighbors; this can also be generalized to mixed graphs.

Contribution. In order to better explore the parameterized complexity of MixedColoring, we first introduce and analyze the generalizations of neighborhood diversity and cliquewidth to mixed graphs, which we call mixed neighborhood diversity (mnd) and mixed cliquewidth (mcw), respectively. Since in MixedColoring we can assume that the input graph does not contain directed cycles, we only analyze these parameters for mixed graphs without directed cycles. We explore their relationships with existing parameters defined on the underlying undirected graph, such as vertex cover (vc), treedepth (td), pathwidth (pw), feedback vertex set (fvs), treewidth (tw), as well as neighborhood diversity (nd) and cliquewidth (cw); see the full version [26] for detailed definitions. Specifically, we show that mcw strictly bounds cw (Proposition 1) and is strictly bounded by tw (Proposition 2). Similarly, we show that mnd strictly bounds nd (Proposition 5) and is strictly bounded by vc (Proposition 6). Furthermore, we show that mnd strictly bounds mcw (Proposition 7), while mcw and nd are incomparable (Proposition 8).

Since arcs enforce precedence constraints on the colors of the vertices, *transitive arcs*, i.e., arcs between vertices connected by a directed path along other arcs, do not influence the properness of colorings of mixed graphs. Adding all transitive arcs (and removing any edges parallel to transitive arcs) results in the *transitive closure* G^+ of a mixed graph G (without directed cycles). We denote with α^+ the *transitive closure counterpart* of the parameter α, i.e., $\alpha^+(G) = \alpha(G^+)$. We show that vc, td, pw, fvs, and tw are strictly bounded by their transitive closure counterparts (Proposition 9), while for mcw and mnd the situation is inverted, with mcw and mnd strictly bounding mcw^+ and mnd^+, respectively (Proposition 12). Meanwhile, cw and nd are incomparable to their transitive closure counterparts (Proposition 13). The resulting hierarchy of parameters is visualized in Figure 1.

Furthermore, we also consider bounds for the *chromatic number* (χ) of mixed graphs, i.e., the minimum number of colors needed for a proper coloring. After providing some simple lower bounds, i.e., that the chromatic number of a mixed graph is lower-bounded by the chromatic number of the underlying undirected graph as well as by *maxrank* (Λ), the length of the longest directed path, we generalize and tighten a bound by Gutowski et al. [18] on the chromatic number of mixed interval graphs (Theorem 15). Based on this, we show that the chromatic number of a mixed graph is bounded by tw^+ as well as td and vc (Corollary 17).

Using the newly established parameter relationships, we analyze the parameterized complexity of MixedColoring in Section 3. The results are summarized in Table 1 and visualized in Figure 1. We conclude with an outlook on open problems in Section 4.

Coloring Directed Graphs. For directed graphs, Neumann-Lara [28] introduced the dichromatic number and the corresponding coloring, where each color class induces an acyclic subgraph. This problem, known as DigraphColoring [21], also generalizes coloring undirected graphs, as each edge can be replaced by two opposite arcs. Therefore, DigraphColoring is NP-hard. In contrast, MixedColoring is easy on directed graphs: if the graph is acyclic, greedily coloring vertices in topological order yields an optimal coloring. Otherwise, the graph cannot be properly colored. This difference arises due to the fact that DigraphColoring treats arcs very differently: in MixedColoring, there cannot be arcs between vertices of the same color, whereas in DigraphColoring there can be arcs between vertices of the same color, as long as these arcs do not form a directed cycle.

On directed acyclic graphs, MixedColoring, especially when considering the transitive closure, behaves somewhat similar to OrientedColoring [31], where, between each pair of color classes, all arcs must have the same orientation. However, contrary to MixedColoring, OrientedColoring does not forbid directed cycles.

Conventions. For positive integer k, we use $[k]$ as shorthand for $\{1, \ldots, k\}$. We say that a mixed graph G is a *partial orientation* of its underlying undirected graph. The full versions of proofs marked with a (clickable) star ($\star$) can be found in the full version of this paper [26].

2 Mixed Graph Parameters

We define the *mixed cliquewidth* of a mixed graph G, denoted by $\mathsf{mcw}(G)$, to be the minimum number of distinct labels needed to construct G using the following operations:

- creation of a new vertex with label i, denoted by i.
- disjoint union of two labeled mixed graphs, denoted by $\oplus$.
- adding an (undirected) edge between every vertex with label i and every vertex with label j, for $i \neq j$, denoted by $\eta_{i,j}$.

- adding a (directed) arc from every vertex with label i to every vertex with label j, for $i \neq j$, denoted by $\alpha_{i,j}$.
- renaming label i to label j, denoted by $\varrho_{i\to j}$.

These are exactly the operations used by Courcelle and Olariu [6] in their definitions of cliquewidth for undirected and directed graphs. Instead of using only $\eta_{i,j}$ for undirected graphs and only $\alpha_{i,j}$ for directed graphs, we allow both operations to be used in order to construct mixed graphs, thereby generalizing the concept of cliquewidth to mixed graphs. We call the sequence of such operations constructing a given mixed graph using ℓ labels a *mixed ℓ-expression*. For example, the directed path of length 4 is constructed by the mixed 3-expression $\alpha_{2,3}(\varrho_{3\to2}(\varrho_{2\to1}(\alpha_{2,3}(\alpha_{1,2}(1 \oplus 2) \oplus 3))) \oplus 3)$. As it turns out, mixed cliquewidth is not bounded by cliquewidth.

▶ **Proposition 1.** *For any mixed graph G (without directed cycles),* $\mathsf{cw}(G) \leqslant \mathsf{mcw}(G)$*. Furthermore,* cw *does not bound* mcw*.*

Proof. The first part follows directly from the definitions, as replacing all arc operations by edge operations in a mixed ℓ-expression constructing G yields an ℓ-expression constructing the underlying undirected graph of G.

Let $\mathrm{AG}(G)$ be the subgraph of G resulting from omitting all edges. It holds that $\mathsf{mcw}(\mathrm{AG}(G)) \leqslant \mathsf{mcw}(G)$, as by omitting the edge operations in a mixed ℓ-expression constructing G we obtain a mixed ℓ-expression for $\mathrm{AG}(G)$. Using this, we construct a family of mixed graphs $(G_\ell)_{\ell\geqslant1}$ with $\mathsf{cw}(G_\ell) = 2$ and $\mathsf{mcw}(G_\ell) \geqslant \ell + 1$. We construct G_ℓ by orienting an $(\ell \times \ell)$-grid graph such that no directed cycles are created. As shown by Golumbic and Rotics [17], the cliquewidth of an $(\ell \times \ell)$-grid graph is $\ell + 1$ and thus $\mathsf{mcw}(G_\ell) \geqslant \mathsf{mcw}(\mathrm{AG}(G_\ell)) \geqslant \mathsf{cw}(\mathrm{AG}(G_\ell)) = \ell + 1$. Furthermore, between any pair of vertices of G_ℓ that are not connected by an arc, we add an edge. This results in the underlying undirected graph of G_ℓ being a complete graph, which has cliquewidth 2, and thus $\mathsf{cw}(G_\ell) = 2$. ◀

Courcelle and Olariu [6] showed that the cliquewidth of undirected as well as of directed graphs is bounded by treewidth. We can utilize their result for directed graphs to show that mixed cliquewidth (on mixed graphs without directed cycles) is also bounded by treewidth. To this end, we transform a mixed graph G without directed cycles into its *corresponding directed graph $D(G)$* by replacing each edge $\{u, v\}$ by two opposite arcs (u, v) and (v, u). Since mixed graphs without directed cycles cannot contain opposite arcs (as they would form a directed cycle of length two) this transformation is injective. We define the *directed cliquewidth* of a mixed graph G without directed cycles, denoted $\mathsf{dcw}(G)$, to be the (directed) cliquewidth of the corresponding directed graph $D(G)$. As we show now, mixed and directed cliquewidth are, while not equal, equivalent under the lens of parameterized complexity.

▶ **Proposition 2** ($\star$)**.** *For any mixed graph G (without directed cycles), it holds that* $\mathsf{dcw}(G) \leqslant \mathsf{mcw}(G) \leqslant 2\,\mathsf{dcw}(G) \leqslant 2(2^{\mathsf{tw}(G)+1} + 1)$*. Further, there exists a mixed graph $\tilde{G}$ such that* $\mathsf{dcw}(\tilde{G}) \neq \mathsf{mcw}(\tilde{G})$*.*

Idea. Converting a mixed ℓ-expression for G into a directed ℓ-expression for $D(G)$ is straightforward, as we can replace each edge operation by two opposite arc operations. The inverse requires a careful manipulation of the ℓ-expression for $D(G)$, to ensure that we can replace pairs of opposite arc operations by edge operations.

The partial orientation $\tilde{G}$ of the star $K_{1,2}$, where the center vertex is connected to one leaf by an edge and to the other leaf by an outgoing arc, has $\mathsf{dcw}(\tilde{G}) = 2$ and $\mathsf{mcw}(\tilde{G}) = 3$. ◀

In order to generalize neighborhood diversity to mixed graphs without directed cycles, we first need to define the different types of neighbors that occur in a mixed graph. Given a vertex v of a mixed graph G, the *outgoing neighbors* of v, denoted by $N^+(v)$, are all vertices u to which v has an *outgoing arc* (v, u). Analogously, the *incoming neighbors* of v, denoted by $N^-(v)$, are all vertices u from which v has an *incoming arc* (u, v). The *undirected neighbors* of v, denoted by $N^{\mathrm{u}}(v)$, are all vertices u which are connected to v by an edge $\{u, v\}$. Thus, the *neighborhood* of a vertex v, denoted by $N(v)$, is the union of its incoming, outgoing, and undirected neighbors.

The parameter neighborhood diversity is defined for undirected graphs via an equivalence relation on the vertices, where two vertices u, v are of the same type, denoted $u \sim_{\mathsf{nd}} v$, if they have the same neighbors, i.e., if $N(u) \setminus \{v\} = N(v) \setminus \{u\}$. We generalize this equivalence relation to mixed graphs by further distinguishing between incoming, outgoing, and undirected neighbors. Therefore, two vertices u, v of a mixed graph G are of the same *(mixed) type*, denoted $u \sim_{\mathsf{mnd}} v$, if they have the same in-, out-, and undirected neighbors, i.e., if $N^{\mathrm{u}}(u) \setminus \{v\} = N^{\mathrm{u}}(v) \setminus \{u\}$, $N^-(u) = N^-(v)$, and $N^+(u) = N^+(v)$.

▶ **Lemma 3** (⋆)**.** *For any mixed graph G (without directed cycles), the relation $\sim_{\mathsf{mnd}}$ is an equivalence relation. Further, vertices of the same type induce an (undirected) clique or an independent set.*

We call the partition of the vertices into equivalence classes under $\sim_{\mathsf{mnd}}$ a *mixed neighborhood partition*. The *mixed neighborhood diversity* of a mixed graph G, denoted $\mathsf{mnd}(G)$, is the number of equivalence classes under $\sim_{\mathsf{mnd}}$. Note that this generalizes the definition of neighborhood diversity for undirected graphs. Furthermore, for mixed graphs without directed cycles, this is also a generalization of the definition of neighborhood diversity for directed graphs by Fernau et al. [11], as in directed graphs without directed cycles there are no opposite arcs. Note further that a mixed neighborhood partition can be obtained easily: it suffices to check, for each pair of vertices, whether they are of the same type.

As is the case for mixed cliquewidth and cliquewidth, the mixed neighborhood diversity can differ arbitrarily from the neighborhood diversity of the underlying undirected graph since each vertex in a directed path has to be of a different type, as we show now.

▶ **Lemma 4.** *Given a mixed graph G (without directed cycles) that contains a directed path of length ℓ, it holds that $\mathsf{mnd}(G) \geqslant \ell + 1$, with each vertex on the path being of a different type.*

Proof. We know from Lemma 3 that there cannot be an arc between two vertices of the same type. Let $\langle v_0, v_1, \ldots, v_\ell \rangle$ be a directed path of length ℓ in G. Suppose that there are two vertices v_i, v_j with $i < j$ that are of the same type. Since $v_{j-1} \in N^-(v_j)$, it follows that $v_{j-1} \in N^-(v_i)$. However, as $i \leqslant j - 1$, it follows that there exists a directed path from v_i to v_{j-1} in G. Together with the arc (v_{j-1}, v_i), this yields a directed cycle, a contradiction to G containing no directed cycles. ◀

▶ **Proposition 5.** *For any mixed graph G (without directed cycles), $\mathsf{nd}(G) \leqslant \mathsf{mnd}(G)$. Furthermore, nd does not bound mnd.*

Proof. It follows directly from the definitions that two vertices of the same type w.r.t. mnd must also be of the same type w.r.t. nd. Indeed, if $u \sim_{\mathsf{mnd}} v$, it holds that

$$N(u) \setminus \{v\} = N^-(u) \cup N^+(u) \cup N^{\mathrm{u}}(u) \setminus \{v\} = N^-(v) \cup N^+(v) \cup N^{\mathrm{u}}(v) \setminus \{u\} = N(v) \setminus \{u\}$$

(as there cannot be any arc between u and v) and therefore $u \sim_{\mathsf{nd}} v$.

The inverse does not hold, as two vertices of the same type w.r.t. nd may differ in their in- or out-neighbors, as the following family $(G_\ell)_{\ell \geqslant 1}$ of graphs shows. The graph G_ℓ is a partial orientation of the complete graph K_ℓ obtained by orienting $\ell - 1$ edges such that the resulting mixed graph contains a directed Hamiltonian path. It holds that $\mathsf{nd}(G_\ell) = 1$ but, due to Lemma 4, $\mathsf{mnd}(G_\ell) \geqslant \ell$. ◀

Lampis [25] showed that neighborhood diversity is bounded by vertex cover, as the type of vertices not contained in the cover is determined uniquely by their neighbors (in the cover), resulting in a bound of $\mathsf{vc} + 2^{\mathsf{vc}}$. Since mixed neighborhood diversity further distinguishes between in-, out-, and undirected neighbors, we obtain the following bound.

▶ **Proposition 6.** *For any mixed graph G (without directed cycles),* $\mathsf{mnd}(G) \leqslant \mathsf{vc}(G) + 4^{\mathsf{vc}(G)}$.

Proof. Let C be a vertex cover of G of size $\mathsf{vc}(G)$. Since each vertex in $V(G) \setminus C$ only contains neighbors in C, there are at most $4^{\mathsf{vc}(G)}$ different types of vertices in $V(G) \setminus C$, as each vertex in C can be an in-, out-, or undirected neighbor, or not a neighbor at all. Further, there can be at most $\mathsf{vc}(G)$ different types of vertices in C, as C contains $\mathsf{vc}(G)$ vertices. Thus, the total number of types is bounded by $\mathsf{vc}(G) + 4^{\mathsf{vc}(G)}$. ◀

If a parameter does not bound nd (cw) on undirected graphs (or vice versa), it also does not bound mnd (mcw) (or vice versa), as it holds that $\mathsf{mnd} = \mathsf{nd}$ ($\mathsf{mcw} = \mathsf{cw}$) on undirected graphs. Lampis [25] showed that nd bounds cw and that tw (and therefore cw) does not bound nd (nd is even incomparable to tw). It follows from this that mcw does not bound mnd. Using a proof analogous to Lampis, we further obtain the following.

▶ **Proposition 7** ($\star$). *For any mixed graph G (without directed cycles),* $\mathsf{mnd}(G)+1 \geqslant \mathsf{mcw}(G)$. *Further,* mcw *does not bound* mnd.

Regarding the relationship between mcw and nd, we already know that mcw does not bound nd (due to cw not bounding nd). However, while nd bounds cw, it does not bound mcw. We show this by reusing the construction that we used to show that cw does not bound mcw (Proposition 1), as detailed in the following.

▶ **Proposition 8** ($\star$). *The parameters* mcw *and* nd *are incomparable.*

Transitive Closure. Recall that an arc (u, v) is transitive if there exists a directed path from u to v using other arcs. The transitive closure G^+ of a mixed graph G is the mixed graph obtained by adding every transitive arc to G (and removing any resulting parallel edges). As transitively closed graphs are a subset of mixed graphs, all bounds between parameters on mixed graphs also apply between their transitive closure counterparts. Further, if a parameter α does not bound a parameter β on undirected graphs, then α^+ does not bound β^+, as it holds that $\alpha^+ = \alpha$ and $\beta^+ = \beta$ on undirected graphs. Thus, it remains to analyze the relationship between the parameters and their transitive closure counterparts.

All parameters considered in this paper, except for neighborhood diversity and cliquewidth, have the well-known property that they bound their value on any subgraph. Since the underlying graph of the transitive closure of a mixed graph is a subgraph of the underlying graph of the original graph, it follows that these parameters are bounded by their transitive closure counterparts.

▶ **Proposition 9** ($\star$). *For every mixed graph G (without directed cycles) and parameter $\alpha \in \{\mathsf{vc}, \mathsf{td}, \mathsf{fvs}, \mathsf{pw}, \mathsf{tw}\}$ it holds that $\alpha(G) \leqslant \alpha^+(G)$. Furthermore, these parameters do not bound their transitive closure counterparts.*

Idea. The second part can be shown using a family of oriented stars, where half of the edges are oriented towards the center vertex and the other half are oriented away from the center vertex. ◀

However, for neighborhood diversity and cliquewidth (and their mixed variants), the situation is different. Contrary to the other parameters, these parameters are not bounded by their transitive closure counterparts, as shown in the following.

▶ **Proposition 10.** *The parameters* mnd*,* nd*,* mcw*, and* cw *are not bounded by their transitive closure counterparts.*

Proof. We first construct a family $(G_\ell)_{\ell \geqslant 1}$ of grid graphs where $2 \geqslant \mathsf{mcw}^+(G_\ell) \geqslant \mathsf{cw}^+(G_\ell)$ but $\mathsf{mcw}(G_\ell) \geqslant \mathsf{cw}(G_\ell) = \ell + 1$. The graph G_ℓ is a partial orientation of the $(\ell \times \ell)$-grid graph where $\ell^2 - 1$ edges are oriented such that G_ℓ contains a directed Hamiltonian path, i.e., a directed path of length $\ell^2 - 1$ containing all vertices. Golumbic and Rotics [17] showed that the cliquewidth of $(\ell \times \ell)$-grid graphs is $\ell + 1$. Together with Proposition 1 we obtain that $\mathsf{mcw}(G_\ell) \geqslant \mathsf{cw}(G_\ell) = \ell + 1$. As G_ℓ contains a directed Hamiltonian path, it follows that G_ℓ^+ is the *acylic tournament graph*, the directed graph with exactly one arc between each pair of vertices (and without directed cycles). Using the following claim, we obtain that $2 \geqslant \mathsf{mcw}^+(G_\ell) \geqslant \mathsf{cw}^+(G_\ell)$.

▷ Claim 11 (⋆). The mixed cliquewidth of an acyclic tournament graph is (at most) 2.

For neighborhood diversity, we construct a separate family $(G'_\ell)_{\ell \geqslant 1}$ based on tripartite graphs. The graph G'_ℓ consists of three independent sets $\{u_1, \ldots, u_\ell\}$, $\{v_1, \ldots, v_\ell\}$, and $\{w_1, \ldots, w_\ell\}$. We add arcs (u_i, v_j) and (v_i, w_j) for $i, j \in [\ell]$, such that every u has an outgoing arc to every v and every v has an outgoing arc to every w. To ensure that vertices of different sets are of different types, we further add three vertices u^*, v^*, and w^*. We then add edges $\{u^*, u_i\}$, $\{v^*, v_i\}$, and $\{w^*, w_i\}$ for all $i \in [\ell]$. Lastly, we add arcs (u_i, w_i) for $i \in [\ell]$. Thus, each vertex u_i has a distinct type, as does every vertex w_i. Meanwhile, all vertices v_i are of the same type. Further, the vertices u^*, v^*, and w^* have distinct types. This results in $\mathsf{mnd}(G'_\ell) = \mathsf{nd}(G'_\ell) = 2\ell + 4$. However, in the transitive closure, each vertex u_i has an outgoing arc to every vertex w_j and therefore all u_i are of the same type and all w_i are of the same type, resulting in $\mathsf{mnd}^+(G'_\ell) = \mathsf{nd}^+(G'_\ell) = 6$. ◀

In fact, both mixed neighborhood diversity and mixed cliquewidth bound their transitive closure counterparts, as shown in the following.

▶ **Proposition 12 (⋆).** *For any mixed graph G (without directed cycles), it holds that* $\mathsf{mnd}^+(G) \leqslant \mathsf{mnd}(G)$ *and* $\mathsf{mcw}^+(G) \leqslant 4^{\mathsf{mcw}(G)} \cdot \mathsf{mcw}(G)$.

Idea. For mixed neighborhood diversity, it holds that two vertices u, v of the same type in G are also of the same type in G^+, as the set of vertices to (from) which u, v have outgoing (incoming) paths in G are equivalent.

For mixed cliquewidth, we show that given a mixed ℓ-expression constructing G, we can obtain a $(4^\ell \cdot \ell)$-expression constructing G^+. The idea is to keep track of the labels of vertices to which a given vertex v has outgoing paths and the labels of vertices from which it has incoming paths. This can be achieved by using 4^ℓ additional labels for each original label. These additional labels then allow us to add the transitive arcs, resulting in the claimed bound. ◀

The same does not hold for neighborhood diversity and cliquewidth, as these are in fact incomparable to their transitive closure counterparts, as the following shows.

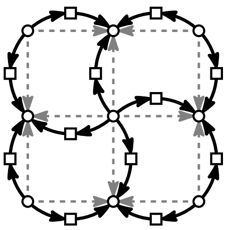

Figure 2 Example of the graph G_ℓ for $\ell = 3$. The grid vertices are depicted as disks and the arc vertices as squares. The grid vertices form an independent set in G_ℓ but induce a (3×3)-grid in G_ℓ^+ (the transitive arcs are depicted as dashed gray). The edges between arc vertices and grid vertices are omitted for visual clarity.

▶ **Proposition 13.** *The parameters* nd *and* cw *are incomparable to their transitive closure counterparts.*

Proof. We already know from Proposition 10 that these parameters are not bounded by their transitive closure counterparts. Thus, it remains to show that they do not bound their transitive closure counterparts. For this, we construct a family of graphs $(G_\ell)_{\ell \geqslant 2}$ such that $\mathsf{nd}(G_\ell) = \mathsf{cw}(G_\ell) = 2$ while $\mathsf{nd}^+(G_\ell) \geqslant \mathsf{cw}^+(G_\ell) \geqslant \ell + 1$.

The graph G_ℓ consists of ℓ^2 *grid vertices* and $2\ell(\ell-1)$ *arc vertices* and is constructed such that the grid vertices form an independent set in G_ℓ but induce an $(\ell \times \ell)$-grid in G_ℓ^+. This is achieved through the usage of the arc vertices, which connect two grid vertices adjacent in G_ℓ^+ with a directed path, thus ensuring that the transitive arc is present in G_ℓ^+. Further, each arc vertex is connected to all grid vertices via edges, except for those to which it is connected via an arc. This ensures that all arc vertices are of the same type and that all grid vertices are of the same type, resulting in $\mathsf{nd}(G_\ell) = \mathsf{cw}(G_\ell) = 2$; see Figure 2 for an example of the construction for $\ell = 3$. As shown by Golumbic and Roctis [17], the cliquewidth of an $(\ell \times \ell)$-grid is $\ell + 1$. Furthermore, it is easy to see that the cliquewidth of a graph bounds the cliquewidth of induced subgraphs. Thus, it follows that $\mathsf{nd}^+(G_\ell) \geqslant \mathsf{cw}^+(G_\ell) \geqslant \ell + 1$. ◀

Chromatic Number. Clearly, any proper coloring of a mixed graph is also a proper coloring of the underlying undirected graph and thus the chromatic number of the underlying undirected graph lower bounds the chromatic number of the mixed graph. Recall that the maxrank (Λ) of a mixed graph is the length of the longest directed path in the graph. For a given vertex v, we define the *inrank*, denoted as $\lambda^-(v)$, as the length of the longest directed path ending in v. Furthermore, as vertices along a directed path must be colored increasingly, every vertex has a color that is at least its inrank. Thus, it holds that the chromatic number bounds the maxrank. We summarize these insights in the following proposition.

▶ **Proposition 14.** *For any mixed graph G (without directed cycles), $\chi_{\mathrm{u}}(G) \leqslant \chi(G)$ and $\Lambda(G) \leqslant \chi(G)$.*

For undirected graphs, several (upper) bounds on the chromatic number are known, such as treewidth and maximum degree. However, these bounds do not generalize to mixed graphs, as a simple directed path of length ℓ has treewidth 1 and maximum degree 2, but chromatic number $\ell + 1$. Gutowski et al. [18, Theorem 9 & Proposition 10] showed that for mixed interval graphs it holds that $\chi \leqslant (\Lambda + 1) \cdot \omega$, where ω is the clique number, and that this bound is asymptotically tight. They achieved this by introducing the concept of *layering*, which is a partition of the vertex set into $\Lambda + 1$ layers $\langle L_0, \ldots, L_\Lambda \rangle$ such that the layer L_i contains all vertices of inrank i. In a layering, each arc is oriented towards the higher layer, as it must hold for every arc (u, v) that $\lambda^-(u) < \lambda^-(v)$. It follows that each subgraph $G[L_i]$ induced by a layer L_i is an undirected graph. Thus, we can iteratively color each layer with disjoint color sets, resulting in a proper coloring of the entire graph as formalized in the following theorem.

▶ **Theorem 15** (⋆). *For any mixed graph G (without directed cycles) and its corresponding layering $\langle L_0, L_1, \ldots, L_\Lambda \rangle$, it holds that $\chi(G) \leqslant \sum_{i=0}^{\Lambda} \chi_u(G[L_i])$. Further, this bound is tight.*

Regarding the transitive closure, note that $\Lambda^+ = \Lambda$ and $\chi^+ = \chi$, as adding transitive arcs neither changes the length of the longest directed path nor the validity of a coloring. Thus, we obtain the following bounds on maxrank.

▶ **Proposition 16** (⋆). *For any mixed graph G (without directed cycles), it holds that $\Lambda(G) + 1 \leqslant \mathsf{mnd}^+(G)$, $\Lambda(G) \leqslant \mathsf{tw}^+(G)$, $\Lambda(G) \leqslant 2^{\mathsf{td}(G)} - 2$, and $\Lambda(G) \leqslant 2\mathsf{vc}(G)$.*

As the chromatic number of a subgraph is a lower bound for the chromatic number of the entire graph, it follows from Theorem 15 and Proposition 16 that any parameter that bounds the chromatic number of undirected graphs as well as the maxrank also bounds the chromatic number of mixed graphs.

▶ **Corollary 17** (⋆). *The chromatic number of mixed graphs is bounded by tw^+ as well as td and vc. In fact, it even holds for every mixed graph G that $\chi(G) \leqslant 2\mathsf{vc}(G) + 1$.*

The resulting hierarchy is visualized in Figure 1.

3 Parameterized Complexity of MixedColoring

Recall that a problem is *fixed-parameter tractable* (FPT) with respect to a parameter k if it can be solved in $f(k) \cdot n^{\mathcal{O}(1)}$ time, where f is a computable function and n is the size of the input. A problem is *slice-wise polynomial* (XP) with respect to k if it can be solved in $\mathcal{O}(n^{f(k)})$ time (where f and n are as above). Clearly, FPT ⊆ XP. Note that, given an FPT- or XP-algorithm for k-(Mixed)Coloring that is parameterized by the number of colors, k, we can easily obtain an FPT- or XP-algorithm parameterized by the chromatic number. Therefore, we state our results with respect to the parameter chromatic number (if the runtime is FPT or XP parameterized by the number of colors).

Courcelle's theorem yields for any problem that can be expressed in monadic second-order logic on graphs an FPT-algorithm parameterized by tree- or even cliquewidth (depending on the logic variant used). Like cliquewidth, monadic second-order logic on graphs extends intuitively to mixed graphs, and expressing MixedColoring as a monadic second-order formula is straightforward. However, to our knowledge, Courcelle's theorem has not yet been entirely proven for mixed graphs, with Arnborg et al. [1] only proving the theorem for treewidth but not for cliquewidth. Still, by converting our mixed graphs into directed graphs as in Proposition 2, we can apply the directed version of Courcelle's theorem to obtain the following result.

▶ **Theorem 18** (⋆). *The problem MixedColoring is FPT parameterized by mixed cliquewidth plus chromatic number.*

While Courcelle's theorem yields an FPT-algorithm for MixedColoring, the resulting algorithm is quite impractical. For example, the runtime contains huge constants depending on the length of the formula, i.e., the number of colors, which cannot be bounded by an elementary function unless $\mathsf{P} = \mathsf{NP}$, as shown by Frick and Grohe [14]. Since polynomials are elementary functions, this also implies that this algorithm is not an XP-algorithm parameterized solely by tree- or cliquewidth.

Another Coloring algorithm can also be generalized to MixedColoring: it is well known [7, Theorem 7.9] that Coloring, given a tree-decomposition of width tw, can be solved in $\mathcal{O}^*(k^{\mathsf{tw}})$ time via a dynamic program. By exchanging the checks for a proper coloring with checks for a proper mixed coloring (i.e., additionally checking that no arc is violated), MixedColoring can be solved within the same time bound, as we now summarize.

▶ **Theorem 19.** *Given a mixed graph and a tree-decomposition of width* tw*, the problem k-MixedColoring can be solved in $\mathcal{O}^*(k^{\mathsf{tw}})$ time.*

As an optimal tree-decomposition can be computed in FPT-time w.r.t. treewidth [2] and the chromatic number is bounded by the number of vertices, this yields FPT and XP algorithms for MixedColoring, as summarized in the following.

▶ **Corollary 20.** *MixedColoring is* FPT *parameterized by treewidth plus chromatic number and* XP *parameterized solely by treewidth.*

Note that this algorithm is an improvement over the XP-algorithm developed by Ries and de Werra [30], which runs in $\mathcal{O}(n^{2\mathsf{tw}+4}m^{\mathsf{tw}+2})$ time, where m is the number of edges and n the number of vertices.

We cannot hope for an FPT-algorithm parameterized by treewidth alone, as we show in the following that MixedColoring is W[1]-hard w.r.t. pathwidth and feedback vertex set, which are parameters that bound treewidth. We achieve this via a parameterized reduction from ListColoring, a generalization of Coloring where each vertex can only be assigned a color from a given list. While Fellows et al. [10] only showed that ListColoring is W[1]-hard w.r.t. treewidth, their reduction actually even yields W[1]-hardness w.r.t. vertex cover, as we show in the following lemma.

▶ **Lemma 21** ($\star$). *ListColoring parameterized by vertex cover is* W[1]*-hard.*

To reduce ListColoring to k-MixedColoring, where k is the total number of distinct colors in the ListColoring instance, we enforce the color list of each vertex by adding adequate edges to newly inserted directed paths of vertex-length k. This yields the following.

▶ **Lemma 22** ($\star$). *There exists a parameterized reduction w.r.t. pathwidth as well as w.r.t. feedback vertex set from ListColoring to MixedColoring.*

Combining the W[1]-hardness of ListColoring with the parameterized reduction to MixedColoring we obtain the following theorem.

▶ **Theorem 23.** *MixedColoring is* W[1]*-hard w.r.t. feedback vertex set as well as w.r.t. pathwidth.*

Recall that Coloring is FPT with respect to neighborhood diversity [15] and XP with respect to cliquewidth [22]. In contrast to that, we now show that MixedColoring is paraNP-hard with respect to both of these parameters, even in more restricted settings.

▶ **Theorem 24** ($\star$). *MixedColoring is* paraNP*-hard w.r.t. mixed cliquewidth plus maxrank.*

Idea. Given a set of strings $\mathcal{S}$ over the binary alphabet $\{0, 1\}$ and a positive integer k, the problem ShortestSuperstring asks whether there is a string of length at most k that is a superstring of each string in $\mathcal{S}$. ShortestSuperstring is NP-hard [29].

Figure 3 (a) The mixed graph resulting from an instance of ShortestSuperstring with strings $\{01, 100, 11\}$ and shortest superstring 1010. (b) The construction resulting from splitting the path corresponding to the string 100 at each vertex and connecting the resulting vertices to distinct cliques to enforce that they receive the same color in a proper k-coloring.

We reduce from ShortestSuperstring by constructing a mixed graph G such that a proper k-coloring of G corresponds to a length-k superstring of $\mathcal{S}$. We achieve this by creating a directed path P_S for each string S in $\mathcal{S}$ such that the vertices of the path correspond to the characters of the string; we add edges such that vertices corresponding to different characters cannot receive the same color. See Figure 3a for an example. Thus, the color of a vertex corresponds to the position of the corresponding character in the superstring, and the directions of the paths enforce that the characters of each string appear in the correct order.

To ensure a constant maxrank, we then split each directed path P_S at every vertex into two vertices, with one vertex obtaining the incoming and the other the outgoing arc. We then connect each new pair of vertices to a distinct $(k-1)$-clique, thus enforcing that the two vertices receive the same color in any proper k-coloring; see Figure 3b. Therefore, regarding proper k-colorings, they can be treated as a single vertex, while the maxrank of the resulting graph is 1, due to all directed paths having been split. Furthermore, the resulting graph has a mixed cliquewidth of at most 6. ◀

▶ **Theorem 25.** *MixedColoring is paraNP-hard parameterized by neighborhood diversity, even if considering the neighborhood diversity of the transitive closure.*

Proof. We reduce from the following problem, which is NP-hard [16, SS9].

Problem:	PrecedenceConstrainedScheduling
Input:	Sets T_1, T_2 of unit-length tasks, a partial order $\prec$ on $T_1 \cup T_2$, an integer D.
Question:	Can the tasks be scheduled with makespan D on two parallel machines M_1 and M_2 such that the tasks in T_1 are assigned to M_1, the tasks in T_2 are assigned to M_2, and all precedence constraints are respected (i.e., if $t \prec t'$, then t must finish before t' starts)?

Given an instance $(T_1, T_2, \prec, D)$ of PrecedenceConstrainedScheduling, we create a $4D$-MixedColoring instance G such that each block of four consecutive colors corresponds to one time unit on the machines.

First, we create a directed path P consisting of $4D$ vertices $\langle p_1, \ldots, p_{4D} \rangle$, which we will use to ensure that each task is scheduled on its assigned machine. Note that in any proper $4D$-coloring, the path P has the same unique coloring, with each vertex p_i receiving the color i. For each task $t \in T_1 \cup T_2$, we create two *task vertices*, the *start vertex* v_t^- and the *end vertex* v_t^+, connected via the arc (v_t^-, v_t^+). We denote with $[i]_k$ the set of integers that are congruent to i modulo k. For each start vertex v_t^- of a task t in T_1, we add an edge to

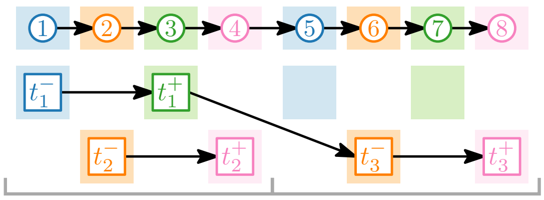

Figure 4 The mixed graph resulting from an instance of PRECEDENCECONSTRAINEDSCHEDULING with $T_1 = \{t_1\}$ and $T_2 = \{t_2, t_3\}$ as well as precedence constraints $t_1 \prec t_3$ and deadline 2. Four consecutive colors correspond to one time unit on the machines, and the colored slots correspond to the colors that the task vertices of the corresponding type can receive. The edges are omitted for visual clarity.

each vertex p_i of the path P with $i \notin [1]_4$. As a result, v_t^- receives a color in $[1]_4$ in every proper $4D$-coloring of G. Analogously, we add edges between each end vertex v_t^+ of a task t in T_1 and each vertex p_i of the path P with $i \notin [3]_4$ to ensure that v_t^+ receives a color in $[3]_4$. For tasks in T_2, we similarly add edges to ensure that the start vertices receive colors in $[2]_4$ and the end vertices receive colors in $[4]_4$. The precedence constraints are enforced by adding arcs between the corresponding tasks, i.e., for each $t, t' \in T_1 \cup T_2$ with $t \prec t'$ we add the arc $(v_t^+, v_{t'}^-)$. Lastly, it remains to ensure that no two tasks are scheduled at the same time on the same machine. We achieve this by adding (undirected) edges such that the underlying undirected graph induced by the task vertices forms a clique. An example of the resulting graph is shown in Figure 4.

▷ Claim 26 (⋆). The resulting graph G has a proper $4D$-coloring if and only if there is a schedule of the tasks on their assigned machines that respects the precedence constraints and finishes by time D.

▷ Claim 27 (⋆). The transitive closure of G has a neighborhood diversity of 8.

As these claims show, the resulting MIXEDCOLORING instance is equivalent to the original PRECEDENCECONSTRAINEDSCHEDULING instance, with the neighborhood diversity of the transitive closure being 8. Thus, MIXEDCOLORING is paraNP-hard w.r.t. neighborhood diversity, even when considering the neighborhood diversity of the transitive closure. ◀

The picture changes if we parameterize by *mixed* neighborhood diversity.

▶ **Theorem 28.** *MIXEDCOLORING is* FPT *parameterized by mixed neighborhood diversity.*

Proof. Let G be a mixed graph. We can assume w.l.o.g. that each type of G induces a clique, as all vertices of a type inducing an independent set can be merged into a single vertex. This is due to the fact that they have the same neighborhoods and therefore can be assigned the same color. Let $\mathcal{S} = \{C_1, \dots, C_{\mathsf{mnd}}\}$ be the set of types of G, where each C_i is a clique in G. In the following, we write $\{C_i, C_j\} \in E(G)$ and $(C_i, C_j) \in A(G)$ to express that there is an undirected edge or a directed arc, respectively, between the types C_i and C_j in G. For a coloring c of G and a set $U \subseteq V(G)$, let $c(U) = \{c(u) : u \in U\}$. If c is proper, then, for every $(C_i, C_j) \in A(G)$, all colors in $c(C_i)$ are smaller than all colors in $c(C_j)$. We define a *type-endpoint preorder* p as a function that assigns, to each type C, two integers $p^-(C), p^+(C) \in [\ell]$ (with $p^-(C) < p^+(C)$ and $\ell \leqslant 2\,\mathsf{mnd}$), which we call the *type endpoints* of C. We say that a coloring c *corresponds* to a type-endpoint preorder p if there is a sequence $\langle c_1, \dots, c_\ell \rangle$ of ℓ ascending (not necessarily consecutive) colors such that, for each type C, the set $c(C)$ is contained in the halfopen interval $[c_{p^-(C)}, c_{p^+(C)})$. We

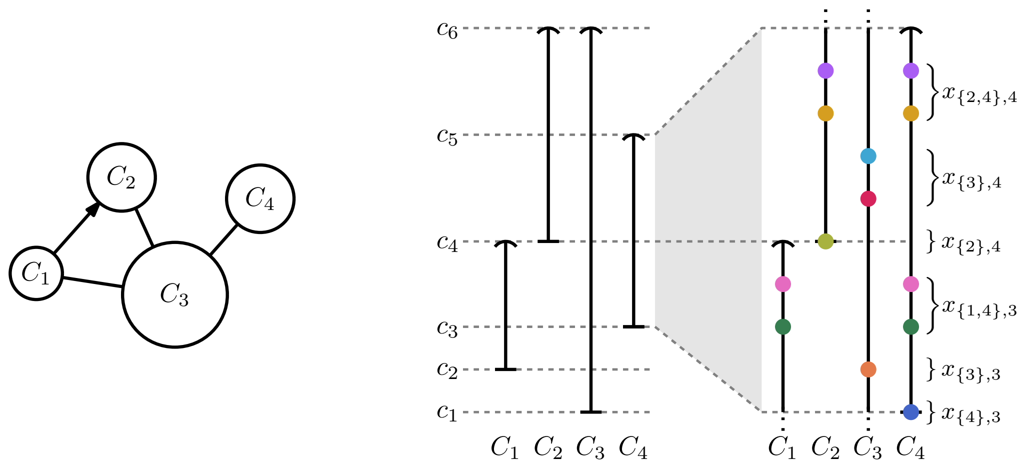

Figure 5 A mixed graph with four types $C_1, \ldots, C_4$ and a preorder with $\ell = 6$ type endpoints. Since it holds for its only arc (C_1, C_2) that $p^+(C_1) = c_4 = p^-(C_2)$, the preorder is proper. On the right, we depict a part of a coloring corresponding to the preorder, together with the variables of the ILP. Note that most of the variables are zero, either due to a constraint of the ILP, such as for $x_{\{1\},4}$ or $x_{\{3,4\},3}$, or simply due to the fact that there are no colors shared exclusively by the given set of types in the given interval, such as for $x_{\{1\},3}$.

say that a (type-endpoint) preorder p is *proper* if, for each arc $(C_i, C_j) \in A(G)$, it holds that $p^+(C_i) \leqslant p^-(C_j)$. An example of a proper preorder and a corresponding coloring is shown in Figure 5.

▷ Claim 29 (⋆). Given a proper (type-endpoint) preorder p, any coloring that corresponds to p does not violate any arcs. Further, any proper coloring corresponds to at least one proper preorder.

In total, there are $2\,\mathsf{mnd}$ type endpoints and thus at most $(2\,\mathsf{mnd})!$ orders of these endpoints. Since type endpoints can be equal, there are at most $(2\,\mathsf{mnd})! \cdot 2^{2\,\mathsf{mnd}} \in 2^{\mathcal{O}(\mathsf{mnd}\log\mathsf{mnd})}$ (proper) preorders. Therefore, we can enumerate all proper preorders in FPT-time w.r.t. mixed neighborhood diversity. We can thus check whether G is k-colorable by checking, for each proper preorder p, whether there exists a proper k-coloring of G that corresponds to p. Since we know that such a coloring does not violate any arcs, it suffices to check whether it does not violate any edges. This problem has similarities to Coloring, differing in that we additionally require the colors within each type to adhere to the given preorder. For Coloring, Ganian [15] provided an FPT-algorithm w.r.t. neighborhood diversity based on an ILP. In the following, we adapt his ILP to additionally respect a given proper preorder p.

Since we only need to decide whether there exists a k-coloring c of G that corresponds to the given proper preorder p, we do not need an objective function. For an overview of the ILP, see the full version [26]. We have two kinds of variables. First, for every $i \in [\ell]$, we introduce an integer variable $c_i \in [k]$ to obtain a sequence $\langle c_1, \ldots, c_\ell \rangle$ of colors. We enforce that these colors form an ascending sequence (by adding, for each $i \in [\ell - 1]$, the constraint $c_i + 1 \leqslant c_{i+1}$). Second, for each $i \in [\ell - 1]$ and each subset $\mathcal{S}' \subseteq \mathcal{S}$, we introduce an integer variable $x_{\mathcal{S}',i} \in \{0, \ldots, k\}$, which represents the number of colors in the interval $[c_i, c_{i+1})$ that are shared exclusively by the types in $\mathcal{S}'$. Formally, we want that $x_{\mathcal{S}',i} = |\{d \in [c_i, c_{i+1}) \mid (\forall C \in \mathcal{S}' \colon d \in c(C)) \wedge (\forall C \in \mathcal{S} \setminus \mathcal{S}' \colon d \notin c(C))\}|$, where $c(C)$ is the set of colors assigned to the vertices of type C by the coloring c; see Figure 5 for an example. For each $i \in [\ell - 1]$, we need to ensure that the ILP does not assign too many colors in

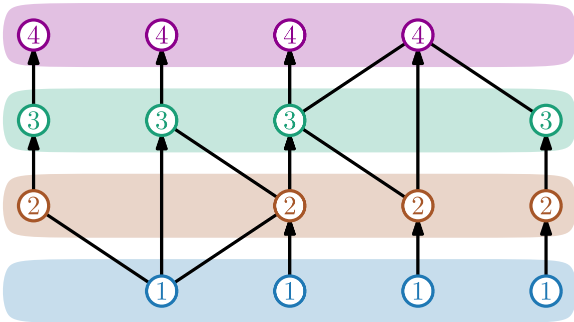

Figure 6 A uniquely 4-colorable mixed graph where no color class is a maximal independent set.

the interval $[c_i, c_{i+1})$. We achieve this by adding the constraint $\sum_{\mathcal{S}' \subseteq \mathcal{S}} x_{\mathcal{S}',i} \leqslant c_{i+1} - c_i$. Moreover, we want the coloring to correspond to the given preorder p. To achieve this, for each type C, we first ensure that no color $d < c_{p^-(C)}$ is assigned to C (by adding the constraint $\sum_{i=1}^{p^-(C)-1} \sum_{\mathcal{S}' \subseteq \mathcal{S}\colon C \in \mathcal{S}'} x_{\mathcal{S}',i} = 0$). Second, we ensure that enough colors are assigned to C in the interval formed by its two type endpoints (by adding the constraint $\sum_{i=p^-(C)}^{p^+(C)-1} \sum_{\mathcal{S}' \subseteq \mathcal{S}\colon C \in \mathcal{S}'} x_{\mathcal{S}',i} = |C|$). Third, we ensure that no color $d \geqslant c_{p^+(C)}$ is assigned to C (by adding the constraint $\sum_{i=p^+(C)}^{\ell-1} \sum_{\mathcal{S}' \subseteq \mathcal{S}\colon C \in \mathcal{S}'} x_{\mathcal{S}',i} = 0$). It remains to ensure that no two types connected by an edge share a color. To this end, for each $i \in [\ell - 1]$ and each set $\mathcal{S}' \subseteq \mathcal{S}$ that contains types C and C' with $\{C, C'\} \in E(G)$, we add the constraint $x_{\mathcal{S}',i} = 0$.

▷ Claim 30 (⋆). For a given proper preorder p, there is a feasible solution to the constructed ILP if and only if there is a proper k-coloring of G that corresponds to p.

It is well known [12, Theorem 9.19] that an ILP can be solved in FPT-time w.r.t. the number of variables. Hence, we can solve the ILP in FPT-time w.r.t. mixed neighborhood diversity as it uses $\mathcal{O}(2^{\mathsf{mnd}} \cdot \mathsf{mnd})$ variables. Therefore, by enumerating all proper preorders and checking for each preorder whether a corresponding proper k-coloring exists, we can decide in FPT-time w.r.t. mixed neighborhood diversity whether G is k-colorable. ◀

Furthermore, while MixedColoring is paraNP-hard w.r.t. neighborhood diversity, it becomes FPT if we use the chromatic number as an additional parameter.

▶ **Theorem 31.** *MixedColoring is* FPT *parameterized by neighborhood diversity plus chromatic number.*

Proof. It was shown by Christofides [3] that there exists an optimal coloring of an (undirected) graph where one color class forms a maximal independent set. However, this property does not hold for mixed graphs; see Figure 6. Fortunately, we can strengthen the property by only considering maximal independent sets of vertices with inrank 0, i.e., vertices without incoming arcs, resulting in the following formula for the chromatic number.

▷ Claim 32 (⋆). For every (non-empty) mixed graph G there exists an optimal coloring of G where the first color class forms a maximal independent set in the subgraph induced by the vertices of inrank 0. Thus, for every mixed graph G it holds that $\chi(G) = 0$ if G is empty and $\chi(G) = \min_I \chi(G - I) + 1$ otherwise, where the minimum is taken over all maximal independent sets I in the subgraph induced by the vertices of inrank 0.

Furthermore, we can bound the number of maximal independent sets in the subgraph induced by vertices of inrank 0 by neighborhood diversity and clique number, as the following shows.

▷ Claim 33 (⋆). For every (non-empty) mixed graph G it holds that the number of maximal independent sets in the subgraph induced by the vertices of inrank 0 is at most $(\omega(G)+1)^{\mathsf{nd}(G)}$.

We implement the recursive formula for the chromatic number as a branching algorithm for MixedColoring as follows: Given a mixed graph G and a number of colors k, we branch on all maximal independent sets I in the subgraph induced by the vertices of inrank 0 and recursively check whether $G - I$ is $(k-1)$-colorable until we either reach the empty graph or run out of colors. Thus, the recursion depth is bounded by k. As in each step we branch into at most $(\omega(G)+1)^{\mathsf{nd}(G)}$ subproblems, this results in an overall runtime of $\mathcal{O}^*((\omega(G)+1)^{\mathsf{nd}(G)\cdot k})$, which is FPT w.r.t. neighborhood diversity plus chromatic number, as $\chi(G) \geqslant \omega(G)$. ◀

4 Open Problems

While we defined mixed neighborhood diversity and mixed cliquewidth only for mixed graphs without directed cycles, these parameters can also be defined for mixed graphs with directed cycles. For mixed cliquewidth the definition remains the same, while for mixed neighborhood diversity the definition can be adapted by integrating the restrictions for opposite arcs introduced by Fernau et al. [11]. The relationships of these more general parameters should be analogous; most proofs can be generalized directly since they do not depend on the requirement that mixed graphs do not contain directed cycles. We remark that it seems likely that Courcelle's theorem extends to (general) mixed cliquewidth, which would facilitate its application to other problems on mixed graphs. Further, we think that it would be interesting to investigate how graph parameters and problem complexities change if we remove all transitive arcs from input graphs. Preliminary computational experiments indicate that MixedColoring algorithms benefit from such a data reduction step. Lastly, while we explored the parameterized complexity of MixedColoring, exact (non-parameterized) algorithms for MixedColoring remain largely unexplored (except for ILP-based approaches).

References

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

2 Hans L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25(6):1305–1317, 1996. doi:10.1137/S0097539793251219.

3 Nicos Christofides. An algorithm for the chromatic number of a graph. *Comput. J.*, 14(1):38–39, 1971. doi:10.1093/COMJNL/14.1.38.

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

5 Bruno Courcelle and Joost Engelfriet. *Graph Structure and Monadic Second-Order Logic – A Language-Theoretic Approach*, volume 138 of *Encyclopedia of Mathematics and Its Applications*. Cambridge University Press, 2012. doi:10.1017/cbo9780511977619.

6 Bruno Courcelle and Stephan Olariu. Upper bounds to the clique width of graphs. *Discret. Appl. Math.*, 101(1–3):77–114, 2000. doi:10.1016/S0166-218X(99)00184-5.

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

8 Peter Damaschke. Parameterized mixed graph coloring. *J. Comb. Optim.*, 38(2):362–374, 2019. doi:10.1007/S10878-019-00388-Z.

9 Dominique de Werra. Restricted coloring models for timetabling. *Discrete Math.*, 165-166:161–170, 1997. doi:10.1016/S0012-365X(96)00208-7.

10 Michael R. Fellows, Fedor V. Fomin, Daniel Lokshtanov, Frances A. Rosamond, Saket Saurabh, Stefan Szeider, and Carsten Thomassen. On the complexity of some colorful problems parameterized by treewidth. *Inf. Comput.*, 209(2):143–153, 2011. doi:10.1016/J.IC.2010.11.026.

11 Henning Fernau, Florent Foucaud, Kevin Mann, Utkarsh Padariya, and Rajath Rao K. N. Parameterizing path partitions. *Theor. Comput. Sci.*, 1028:115029, 2025. doi:10.1016/J.TCS.2024.115029.

12 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006. doi:10.1007/3-540-29953-X.

13 Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, and Saket Saurabh. Intractability of clique-width parameterizations. *SIAM J. Comput.*, 39(5):1941–1956, 2010. doi:10.1137/080742270.

14 Markus Frick and Martin Grohe. The complexity of first-order and monadic second-order logic revisited. *Ann. Pure Appl. Log.*, 130(1-3):3–31, 2004. doi:10.1016/J.APAL.2004.01.007.

15 Robert Ganian. Using neighborhood diversity to solve hard problems. ArXiv report, 2012. arXiv:1201.3091.

16 Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

17 Martin Charles Golumbic and Udi Rotics. On the clique-width of some perfect graph classes. *Int. J. Foundat. Comput. Sci.*, 11(3):423–443. doi:10.1142/S0129054100000260.

18 Grzegorz Gutowski, Konstanty Junosza-Szaniawski, Felix Klesen, Paweł Rzążewski, Alexander Wolff, and Johannes Zink. Coloring and recognizing mixed interval graphs. In Satoru Iwata and Naonori Kakimura, editors, *34th International Symposium on Algorithms and Computation (ISAAC)*, volume 283 of *LIPIcs*, pages 36:1–36:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.ISAAC.2023.36.

19 Grzegorz Gutowski, Florian Mittelstädt, Ignaz Rutter, Joachim Spoerhase, Alexander Wolff, and Johannes Zink. Coloring mixed and directional interval graphs. In Patrizio Angelini and Reinhard von Hanxleden, editors, *30th International Symposium on Graph Drawing and Network Visualization (GD)*, volume 13764 of *LNCS*, pages 418–431. Springer, 2022. doi:10.1007/978-3-031-22203-0_30.

20 Pierre Hansen, Julio Kuplinsky, and Dominique de Werra. Mixed graph colorings. *Math. Methods Oper. Res.*, 45(1):145–160, 1997. doi:10.1007/BF01194253.

21 Ararat Harutyunyan, Michael Lampis, and Nikolaos Melissinos. Digraph Coloring and Distance to Acyclicity. *Theory Comput. Syst.*, 68(4):986–1013, 2024. doi:10.1007/s00224-022-10103-x.

22 Daniel Kobler and Udi Rotics. Edge dominating set and colorings on graphs with fixed clique-width. *Discrete Appl. Math.*, 126(2–3):197–221, 2003. doi:10.1016/S0166-218X(02)00198-1.

23 Marek Kubale. Interval vertex-coloring of a graph with forbidden colors. *Discrete Math.*, 74(1–2):125–136, 1989. doi:10.1016/0012-365X(89)90204-5.

24 Marek Kubale, editor. *Graph Colorings*, volume 352 of *Contemporary Mathematics*. AMS, Providence, RI, 2004. doi:10.1090/conm/352.

25 Michael Lampis. Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica*, 64(1):19–37. doi:10.1007/s00453-011-9554-x.

26 Antonio Lauerbach, Konstanty Junosza-Szaniawski, Marie Diana Sieper, and Alexander Wolff. The parameterized complexity of coloring mixed graphs. ArXiv report, 2026. arXiv:2604.15274.
27 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Known algorithms on graphs of bounded treewidth are probably optimal. *ACM Trans. Algorithms*, 14(2):13:1–13:30, 2018. doi:10.1145/3170442.
28 Víctor Neumann-Lara. The dichromatic number of a digraph. *J. Combin. Theory Ser. B*, 33(3):265–270, 1982. doi:10.1016/0095-8956(82)90046-6.
29 Kari-Jouko Räihä and Esko Ukkonen. The shortest common supersequence problem over binary alphabet is NP-complete. *Theor. Comput. Sci.*, 16:187–198, 1981. doi:10.1016/0304-3975(81)90075-X.
30 Bernard Ries and Dominique de Werra. On two coloring problems in mixed graphs. *Eur. J. Comb.*, 29(3):712–725, 2008. doi:10.1016/J.EJC.2007.03.006.
31 Éric Sopena. Homomorphisms and colourings of oriented graphs: An updated survey. *Discrete Math.*, 339(7):1993–2005, 2016. doi:10.1016/j.disc.2015.03.018.
32 Yuri N. Sotskov. Mixed graph colorings: A historical review. *Math.*, 8(3):385:1–24, 2020. doi:10.3390/math8030385.