



HAL
open science

Rankwidth meets stability

Patrice Ossona de Mendez, Jaroslav Nešetřil, Michal Pilipczuk, Roman Rabinovich, Sebastian Siebertz

► **To cite this version:**

Patrice Ossona de Mendez, Jaroslav Nešetřil, Michal Pilipczuk, Roman Rabinovich, Sebastian Siebertz. Rankwidth meets stability. Symposium on Discrete Algorithms (SODA), Jan 2021, Alexandria, United States. hal-03373553

HAL Id: hal-03373553

<https://hal.science/hal-03373553>

Submitted on 11 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Rankwidth meets stability

Jaroslav Nešetřil

Charles University (IUUK), Praha, Czech Republic

nesetril@iuuk.mff.cuni.cz

Patrice Ossona de Mendez

CAMS (CNRS, UMR 8557), Paris, France

pom@ehess.fr

Michał Pilipczuk

University of Warsaw, Poland

michal.pilipczuk@mimuw.edu.pl

Roman Rabinovich

Technical University of Berlin, Germany

roman.rabinovich@tu-berlin.de

Sebastian Siebertz

University of Bremen, Germany

siebertz@uni-bremen.de

Abstract

We study two notions of being well-structured for classes of graphs that are inspired by classic model theory. A class of graphs \mathcal{C} is *monadically stable* if it is impossible to define arbitrarily long linear orders in vertex-colored graphs from \mathcal{C} using a fixed first-order formula. Similarly, *monadic dependence* corresponds to the impossibility of defining all graphs in this way. Examples of monadically stable graph classes are nowhere dense classes, which provide a robust theory of sparsity. Examples of monadically dependent classes are classes of bounded rankwidth (or equivalently, bounded cliquewidth), which can be seen as a dense analog of classes of bounded treewidth. Thus, monadic stability and monadic dependence extend classical structural notions for graphs by viewing them in a wider, model-theoretical context. We explore this emerging theory by proving the following:

- A class of graphs \mathcal{C} is a first-order transduction of a class with bounded treewidth if and only if \mathcal{C} has bounded rankwidth and a *stable edge relation* (i.e. graphs from \mathcal{C} exclude some half-graph as a semi-induced subgraph).
- If a class of graphs \mathcal{C} is monadically dependent and not monadically stable, then \mathcal{C} has in fact an unstable edge relation.

As a consequence, we show that classes with bounded rankwidth excluding some half-graph as a semi-induced subgraph are linearly χ -bounded. Our proofs are effective and lead to polynomial time algorithms.

This paper is a part of projects that have received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreements No 810115 – DYNASNET, and No 677651 – TOTAL).



1 Introduction

The search for efficient algorithms has led to the study of the structural properties of graph classes defined by the exclusion of specific substructures. For example, the structure theorem for graphs with excluded minors [38] and for graphs with excluded topological minors [23, 11] formed the basis of many structural and algorithmic studies. A fundamental contribution of these studies was to unveil the particular importance of classes with bounded treewidth, which was confirmed by their specific algorithmic properties. Precisely, Courcelle’s theorem asserts that in classes with bounded treewidth, every property definable in monadic second-order logic (MSO) can be tested efficiently [8].

Based on the exclusion of *shallow* minors (or shallow topological minors), two of the authors proposed a framework for the structural study of classes of sparse graphs, namely *bounded expansion* classes and (more generally) *nowhere dense* classes [33]. This last notion of sparsity is characteristic to monotone classes of graphs with fixed parameter tractable first-order model checking [13, 22].

Much effort has been taken to extend the numerous algorithmic applications of sparse graph classes, in particular of treewidth, to dense graphs. For example, Courcelle’s theorem was extended to classes of bounded *cliquewidth* [10] (or equivalently of bounded *rankwidth* or bounded *NLC-width*), which is the dense analog of treewidth.

The move from sparse to dense is naturally followed by a move from *monotone* classes (i.e. classes closed under subgraphs) to *hereditary* classes (i.e. classes closed under induced subgraphs). Still, strong algorithmic properties are known to emerge when one considers hereditary classes of graphs defined by forbidding simple induced subgraphs (as witnessed by the class of cographs, circle graphs, or perfect graphs), or semi-induced bipartite subgraphs. Recall that a bipartite graph H is a *semi-induced subgraph* of a graph G if there exist two disjoint subsets of vertices A and B of G such that H is isomorphic to the subgraph of G with vertex set $A \cup B$ and all the edges present in G between A and B .

For example, the *VC-dimension* of a graph is defined from the maximum size of a semi-induced subgraph isomorphic to a *powerset graph*, that is, to a bipartite graph with vertex set $U \cup \mathcal{P}(U)$ and edge set $\{xX : x \in U, X \in \mathcal{P}(U), \text{ and } x \in X\}$. Classes with bounded VC-dimension are known to have specific statistical properties, which are at the heart of computational learning theory [3] and of numerous results in algorithms in geometric graph theory (see e.g. [6, 31]).

A stronger assumption is that a graph excludes, as a semi-induced subgraph, some *half-graph*: a bipartite graphs with vertex set $\{a_1, \dots, a_n\} \cup \{b_1, \dots, b_n\}$ and edge set $\{a_i b_j : 1 \leq i \leq j \leq n\}$. It has been observed that half-graphs provide a primary example why irregular pairs cannot be avoided in the statement of Szemerédi’s Regularity Lemma. Indeed, Malliaris and Shelah showed that forbidding a half-graph as a semi-induced subgraph indeed makes it possible to get rid of irregular pairs [30].

In the language of model theory, a class excluding some powerset graph as a semi-induced subgraph (that is, a class with bounded VC-dimension) is said to have a *dependent edge relation*, and a class excluding some half-graph as a semi-induced subgraph is said to have a *stable edge relation* (or to have bounded *order dimension*). This corresponds to the two main dividing lines used in model theory: dependence and stability. In our setting, a class of graphs is *dependent* if every binary relation that is (first-order) definable in it, seen as an edge relation, is dependent. Similarly, a class is *stable* if every definable binary relation is stable. Stronger model theoretical notions are the notions of *monadic dependence* and *monadic stability*, where we restrict binary relations definable not only in graphs from the class in question, but also in all their vertex-colorings. A surprising connection with structural graph theory is that, for a monotone class of graphs, the properties of dependence, monadic dependence, stability, monadic stability, and nowhere-denseness are equivalent [1]. However, without the assumption of monotonicity, the notions of monadic dependence and monadic stability do not collapse and present much wider concepts of well-structuredness than nowhere denseness, and they are suited for the treatment of dense graphs as well. For instance, every class of bounded cliquewidth is monadically dependent [24], but not necessarily monadically stable.

One of our prime motivations is to extend the techniques designed for classes of sparse graphs (i.e. bounded expansion or nowhere dense classes) to the dense setting. For this, it is natural to consider hereditary classes of graphs that are dependent, monadically dependent, stable, or even monadically stable. As recently shown by Fabiański et al. [15], these structural assumptions may be used in a novel way in the design of parameterized algorithms.

Monadic dependence and monadic stability can be also defined using transductions. A (*first-order*) *transduction* is a way to construct target graphs from vertex-colorings of source graphs by fixed first-order formulas (see Section 2 for formal definitions). In this setting, a class is monadically dependent if it has no transduction onto the class of all powerset graphs (equivalently, onto the class of all graphs). It is monadically stable if it has no transduction onto the class of all half-graphs [4]. From a dual point of view, classes with bounded rankwidth are exactly those that are transductions of the class of trivially perfect graphs (equivalently, of the class of tree-orders). Similarly, classes with bounded linear rankwidth are exactly those that are transductions of the class of half-graphs (equivalently, of the class of linear orders) [7].

In this way, transductions form a basic containment notion for graphs, which can be used to define structural properties through forbidding obstructions, similarly to (shallow) minors or (induced) subgraphs. The difference is that transductions represent containment understood in model-theoretical terms, and thus are suited for considering questions related to first-order logic. As the notions of monadic stability and monadic dependence are preserved by taking transductions and they correspond to major dividing lines in model theory, we expect them to be central in the emerging theory.

In order to explore this theory, it is imperative to understand classical concepts of structural graph theory through the lense of transductions. That is, we wish to describe the closures of classes that are known to be well-structured under transductions. This was done e.g. for classes of bounded degree [16] and for classes of bounded expansion [18]. More importantly for this work, in a previous paper, the following characterization of monadically stable classes of bounded linear rankwidth was given.

Theorem 1.1 ([34]). *If a class of graphs \mathcal{C} has bounded linear rankwidth, then the following conditions are equivalent:*

1. \mathcal{C} has a stable edge relation;
2. \mathcal{C} is stable;
3. \mathcal{C} is monadically stable;
4. \mathcal{C} is a transduction of a class with bounded pathwidth.

Conceptually, this result means that if a class of graphs \mathcal{C} has bounded linear rankwidth and excludes some half-graph as a semi-induced subgraph, then graphs from \mathcal{C} can be “sparsified” in the following sense: for each $G \in \mathcal{C}$ we can find a vertex-colored graph G' of bounded pathwidth such that G can be defined from G' using fixed first-order formulas. The much more difficult question whether a result analogous to Theorem 1.1 holds for classes of bounded rankwidth (instead of *linear* rankwidth) could not be answered in [34] and was stated there as a conjecture.

A by-product of the results of [34] is the conclusion that classes of bounded linear rankwidth are linearly χ -bounded. Here, a hereditary class \mathcal{C} of graphs is (linearly) χ -bounded if the chromatic number of graphs in \mathcal{C} is functionally (linearly) bounded by their clique number. This concept was introduced by Gyárfás [26] and has received a lot of attention (see e.g. the surveys [39, 40]).

Our contribution. In this work we prove the conjecture stated in [34] and establish the following:

Theorem 1.2. *If a class of graphs \mathcal{C} has bounded rankwidth, then the following conditions are equivalent:*

1. \mathcal{C} has a stable edge relation;
2. \mathcal{C} is stable;
3. \mathcal{C} is monadically stable;
4. \mathcal{C} is a transduction of a class with bounded treewidth.

The implications $4 \Rightarrow 3 \Rightarrow 2 \Rightarrow 1$ are obvious. For the implication $1 \Rightarrow 4$, we combine the approach presented in [34] with the techniques used by Bonamy and the third author in [5] to prove that classes of bounded rankwidth are polynomially χ -bounded. Using the tree variant of Simon’s factorization due to Colcombet [7], the authors of [5] introduce a bounded-depth recursive decomposition of the tree encoding of a graph of rankwidth at most k into factors, so that the quotient trees satisfy certain Ramsey properties. We show that in the absence of a half-graph, these properties imply that each root-to-leaf path in a quotient tree can be partitioned into a bounded number of blocks, and the only “points of interest” on the paths are borders between consecutive blocks. This leads to an encoding of the graph in question in a graph of bounded treewidth, which can be decoded using fixed first-order formulas. We stress that this encoding/decoding scheme is by no means straightforward: it requires new combinatorial insights and a careful analysis. The proof is constructive and can be implemented as a polynomial time algorithm.

Further, we show that the equivalence of the first three conditions of [Theorem 1.1](#) is in fact a more general phenomenon that occurs in every monadically dependent graph class. Precisely, we prove:

Theorem 1.3. *For a monadically dependent graph class \mathcal{C} , the following conditions are equivalent:*

1. \mathcal{C} has a stable edge relation;
2. \mathcal{C} is stable;
3. \mathcal{C} is monadically stable.

Note that implications $3 \Rightarrow 2 \Rightarrow 1$ are obvious. However, these implications can be strict for dependent but not monadically dependent classes. For the implication $3 \Rightarrow 2$ this is witnessed by the class of 1-subdivided half-graphs, which is dependent and excludes some half-graph as a semi-induced subgraph, but is not monadically stable. For implication $2 \Rightarrow 1$ this is witnessed by the class of 1-subdivided cliques, which is stable and thus dependent, but is not monadically stable.

The proof of implication $1 \Rightarrow 3$ relies on the idea of quantifier elimination. Assuming that \mathcal{C} is not monadically stable, we start with a formula $\varphi(\bar{x}, \bar{y})$ that is unstable in some *monadic expansion* \mathcal{C}^+ of \mathcal{C} ; that is, \mathcal{C}^+ consists of graphs from \mathcal{C} with some unary predicates added. Then we iteratively reduce φ to simpler and simpler unstable formulas while enriching \mathcal{C}^+ with more unary predicates. Eventually we find an atomic formula that is unstable on some monadic expansion of \mathcal{C} , so \mathcal{C} has an unstable edge relation. The assumption that \mathcal{C} is monadically dependent is crucially used in each quantifier elimination step.

Moreover, [Theorem 1.2](#) has important corollaries for classes with *low rankwidth covers/colorings* (introduced in [29]). It follows from [5] that classes with low rankwidth covers are polynomially χ -bounded. Excluding a semi-induced half-graph allows us to get a stronger property.

Theorem 1.4. *Every class with low rankwidth covers and stable edge relation is linearly χ -bounded.*

In particular, [Theorem 1.4](#) implies that classes with bounded rankwidth and stable edge relation are linearly χ -bounded. Also, requiring that a class has a stable edge relation gives the following collapse.

Theorem 1.5. *A class has low rankwidth covers and a stable edge relation if and only if it is a transduction of a class with bounded expansion.*

Our results together with observations present in the literature are illustrated by the semi-lattice of properties of graph classes in [Figure 1](#). See [Figure 5](#) in [Section 6](#) for an extended version of the schema.

2 Preliminaries

Graphs. If k is a positive integer, we write $[k]$ for the set $\{1, \dots, k\}$. We consider finite, simple, undirected graphs. For a graph G we write $V(G)$ for its vertex set and $E(G)$ for its edge set.

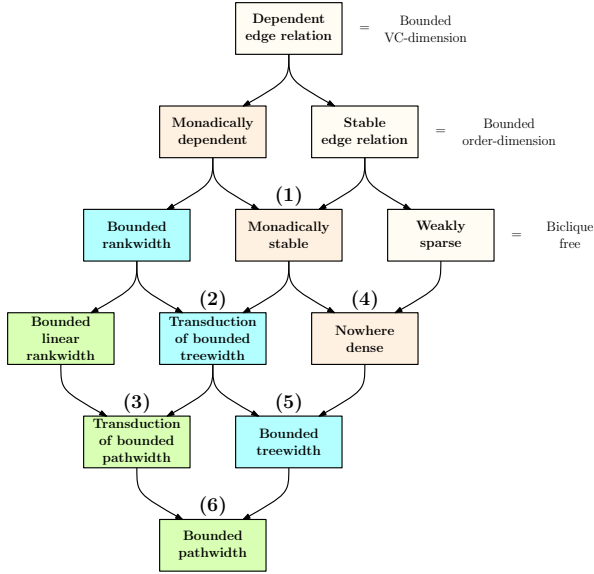


Figure 1: The semi-lattice of property inclusions.

- (1) Monadically stable = monadically dependent \cap stable edge relation (Theorem 1.3);
- (2) Structurally bounded treewidth = bounded rankwidth \cap monadically stable (Theorem 1.2);
- (3) Structurally bounded pathwidth = bounded linear rankwidth \cap structurally bounded treewidth (follows from Theorem 1.1, proved in [34]);
- (4) Nowhere dense = monadically stable \cap weakly sparse (follows from [12], cf. [35]);
- (5) Bounded treewidth = structurally bounded treewidth \cap nowhere dense (= bounded rankwidth \cap weakly sparse [25]);
- (6) Bounded pathwidth = structurally bounded pathwidth \cap bounded treewidth (= bounded linear rankwidth \cap weakly sparse [25]).

A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For $X \subseteq V(G)$, we write $G[X]$ for the subgraph of G induced by X , that is, the subgraph with vertex set X and all edges from G with both endpoints in X . A graph H is an *induced subgraph* of G if there exists $X \subseteq V(G)$ such that H is isomorphic to $G[X]$. For disjoint subsets X, Y of $V(G)$, we write $G[X, Y]$ for the subgraph of G *semi-induced* by X and Y , that is, the subgraph with vertex set $X \cup Y$ and all the edges of G with one endpoint in X and one endpoint in Y . A bipartite graph H is a *semi-induced subgraph* of G if H is isomorphic to $G[X, Y]$ for some disjoint subsets X and Y of $V(G)$. A class \mathcal{C} of graphs excludes a bipartite graph H as a semi-induced subgraph if no $G \in \mathcal{C}$ contains H as a semi-induced subgraph.

The complete bipartite graph (biclique) with each side of size t is denoted by $K_{t,t}$. The *half-graph* of order t is the bipartite graph with vertices $a_1, \dots, a_t, b_1, \dots, b_t$ and edges $a_i b_j$ for all $i, j \in [t]$ with $i \leq j$.

First-order transductions. We assume familiarity with first-order logic and refer to [27] for background. We represent graphs as relational structures over a vocabulary consisting of one binary edge relation symbol E . For a finite set of unary relation symbols Σ , a Σ -*expansion* of a graph G is a structure G^+ obtained from G by adding unary relations with symbols in Σ ; thus, one can think of G^+ as of G with a coloring on the vertex set. If we do not wish to specify Σ , we may simply speak about a *monadic expansion* of G . For a class \mathcal{C} of graphs, a class \mathcal{C}^+ is a *monadic expansion* of \mathcal{C} if there is a finite set of unary relation symbols Σ such that every element of \mathcal{C}^+ is a Σ -expansion of a graph in \mathcal{C} .

For a formula $\varphi(\bar{x})$ in the vocabulary of Σ -expanded graphs, where \bar{x} denotes a tuple of free variables, and a Σ -expanded graph G , we define $\varphi(G) := \{\bar{u} \in V(G)^{|\bar{x}|} : G \models \varphi(\bar{u})\}$. In particular, if A is a unary relation symbol, then $A(G) = \{u \in V(G) : G \models A(u)\}$ and, as expected, $E(G) = \{(u, v) \in V(G) \times V(G) : G \models E(u, v)\}$.

A *simple interpretation* l of graphs in Σ -expanded graphs is a pair $(\nu(x), \eta(x, y))$ consisting of two formulas (in the vocabulary of Σ -expanded graphs), where η is anti-reflexive and symmetric (i.e. $\vdash \neg\eta(x, x)$ and $\vdash \eta(x, y) \leftrightarrow \eta(y, x)$). If G^+ is a Σ -expanded graph, then $H = \mathsf{l}(G^+)$ is the graph with vertex set $\nu(G^+)$ and edge set $\eta(G^+) \cap (\nu(G^+) \times \nu(G^+))$.

A *transduction* T (from graphs to graphs) is a pair $(\Sigma_{\mathsf{T}}, \mathsf{l}_{\mathsf{T}})$, where Σ_{T} is a finite set of unary relation symbols and l_{T} is a simple interpretation of graphs in Σ_{T} -expanded graphs. A graph H can be T -*transduced* from a graph G if there exists a Σ_{T} -expansion G^+ of G such that $\mathsf{l}_{\mathsf{T}}(G^+) = H$. A class \mathcal{D} of graphs can be T -*transduced* from a class \mathcal{C} of graphs if for every graph $H \in \mathcal{D}$ there exists a graph $G \in \mathcal{C}$ such that H can be T -transduced from G . We also say that T is a *transduction from* \mathcal{C}

onto \mathcal{D} . Note that if a class \mathcal{D} can be T-transduced from a class \mathcal{C} and $\mathcal{D}' \subseteq \mathcal{D}$, then also \mathcal{D}' can be T-transduced from \mathcal{C} . A class \mathcal{D} of graphs can be *transduced* from a class \mathcal{C} of graphs if it can be T-transduced from \mathcal{C} for some transduction T. Note that transductions compose in the following sense: If a class \mathcal{D} can be transduced from a class \mathcal{C} and a class \mathcal{E} can be transduced from \mathcal{D} , then \mathcal{E} can be transduced from \mathcal{C} .

Remark 2.1. *A class has bounded rankwidth if and only if it can be transduced from the class of trivially perfect graphs (i.e. from tree-orders) [7]. Hence, if a class \mathcal{D} can be transduced from a class \mathcal{C} of bounded rankwidth, then \mathcal{D} has bounded rankwidth.*

Stability and dependence. A formula $\varphi(\bar{x}, \bar{y})$ is *unstable* on a class \mathcal{C} if for every integer $n \geq 1$ there exists $G \in \mathcal{C}$, $\bar{a}^1, \dots, \bar{a}^n \in V(G)^{|\bar{x}|}$ and $\bar{b}^1, \dots, \bar{b}^n \in V(G)^{|\bar{y}|}$ such that $G^+ \models \varphi(\bar{a}^i, \bar{b}^j)$ if and only if $i \leq j$. The formula $\varphi(\bar{x}, \bar{y})$ is *stable* on \mathcal{C} if it is not unstable on \mathcal{C} . The class \mathcal{C} has a *stable edge relation* if the formula $E(x, y)$ is stable on \mathcal{C} . The class \mathcal{C} is *stable* if every formula $\varphi(\bar{x}, \bar{y})$ is stable on \mathcal{C} . The class \mathcal{C} is *monadically stable* if every monadic expansion \mathcal{C}^+ of \mathcal{C} is stable.

Similarly, a formula $\varphi(\bar{x}, \bar{y})$ is *independent* on a class \mathcal{C} if for every integer $n \geq 1$ there exists $G \in \mathcal{C}$, $\bar{a}^1, \dots, \bar{a}^n \in V(G)^{|\bar{x}|}$ and $\bar{b}^J \in V(G)^{|\bar{y}|}$ for all $J \subseteq [n]$ such that $G^+ \models \varphi(\bar{a}^i, \bar{b}^J)$ if and only if $i \in J$. The formula $\varphi(\bar{x}, \bar{y})$ is *dependent* on \mathcal{C} if it is not independent on \mathcal{C} . The class \mathcal{C} has a *dependent edge relation* if the formula $E(x, y)$ is dependent on \mathcal{C} . The class \mathcal{C} is *dependent* if every formula $\varphi(\bar{x}, \bar{y})$ is dependent on \mathcal{C} . The class \mathcal{C} is *monadically dependent* if every monadic expansion \mathcal{C}^+ of \mathcal{C} is dependent.

It turns out that monadic expansions allow us to circumvent the use of tuples of variables \bar{x} and \bar{y} with length greater than 1, as stated next.

Theorem 2.1 (follows from [4], see also [2]). *A class \mathcal{C} is monadically dependent if and only if there is no transduction from \mathcal{C} onto the class of all finite graphs.*

Theorem 2.2 ([4]). *A class \mathcal{C} is monadically stable if and only if there is no transduction from \mathcal{C} onto the class the all finite half-graphs.*

3 Rankwidth meets stability

In this section we prove [Theorem 1.2](#). We start with some preliminaries on the toolbox introduced by Bonamy and the third author [5], and then proceed to the proper proof.

3.1 The toolbox

Trees. A *tree* is a connected acyclic graph. A *rooted tree* is a tree T with a distinguished node called the *root* of T , denoted $\text{top}(T)$. A rooted tree T defines a partial order on vertices and edges, which we denote by \preceq_T or by \preceq if T is clear from the context. In this partial order we have $\alpha \preceq \beta$ (with $\alpha, \beta \in V(T) \cup E(T)$) if every path in T that starts at the root and includes β also includes α . If α and β are nodes and $\alpha \preceq \beta$, then we also say that α is an *ancestor* of β and β is a *descendant* of α ; note that each node is considered also an ancestor of itself. We also use terms *parent* and *child* with the standard meaning. The parent of a node v of a rooted tree (or the node v itself if v is the root) is denoted by v^\uparrow ; we also denote $(v^\uparrow)^\uparrow$ by $v^{\uparrow\uparrow}$. Note that the ancestor partial order is an *inf-semilattice*, with the meet operation \wedge being the least common ancestor. The *leaves* of a rooted tree T are the \preceq -maximal nodes of T ; the set of all leaves of T is denoted by $L(T)$. Note that from the perspective of first-order logic, a partial order is a transitively oriented comparability graph. In particular, a tree-order is a trivially perfect graph with a transitive orientation.

\mathcal{S}_k -trees. For a positive integer k we let \mathcal{S}_k be the semigroup of all functions from $[k]$ to $[k]$ with composition as the semigroup operation. That is, for $f, g \in \mathcal{S}_k$ we write $f \circ g \in \mathcal{S}_k$ for the function that maps every $i \in [k]$ to $f(g(i))$. An element f of a semigroup is *idempotent* if $f \circ f = f$. An \mathcal{S}_k -tree is a tuple (T, U, ρ, π) , where T is a rooted tree, U is a set, $\rho: E(T) \rightarrow \mathcal{S}_k$ is a labeling of the edges of T by elements of \mathcal{S}_k , and $\pi: U \rightarrow V(T)$ is a mapping from U to the nodes of T .

Rankwidth, cliquewidth and NLC-width. There are various equivalent ways of capturing the treelike structure of dense graphs via hierarchical decompositions. The best known measures are probably *rankwidth* [36], *cliquewidth* [9], and *NLC-width* [41]. All of these measures are equivalent in the sense that if one measure is bounded on a class of graphs, then the other measures are also bounded [28, 36]. We are going to work with the following variant of NLC-width, which is easily seen to be equivalent (in the above sense) to the original definition of NLC-width.

Let (T, U, ρ, π) be an \mathcal{S}_k -tree. For $x, y \in V(T)$ with $x \preceq_T y$ we denote by $\text{path}_T(y, x) = (e_1, \dots, e_s)$ the sequence of edges on the unique path in T from y to x . For $v \in U$ and $x \preceq_T \pi(v)$ we further define $\text{path}_T(v, x) := \text{path}_T(\pi(v), x)$. We implicitly extend ρ to sequences of edges as follows: $\rho((e_1, \dots, e_s)) := \rho(e_s) \circ \dots \circ \rho(e_1)$.

Definition 3.1. Let k be a positive integer and let U be a set. A k -NLC-tree on U is a tuple $\mathfrak{T} = (T, U, \rho, \pi, \eta, \chi)$, where (T, U, ρ, π) is an \mathcal{S}_k -tree, $\eta: V(T) \rightarrow 2^{[k] \times [k]}$ and $\chi: U \rightarrow [k]$. We assume that η is symmetric: for all $x \in V(T)$ and $(i, j) \in [k] \times [k]$, we have $(i, j) \in \eta(x)$ if and only if $(j, i) \in \eta(x)$.

Let $\mathfrak{T} = (T, U, \rho, \pi, \eta, \chi)$ be a k -NLC-tree. We define the *color* in T of $v \in U$ at a node $x \preceq_T \pi(v)$ of T as $\kappa_{\mathfrak{T}}(v, x) := \rho(\text{path}(v, x))(\chi(v))$. The k -NLC-tree \mathfrak{T} generates the graph $G_{\mathfrak{T}}$ with vertex set U , defined as follows: For $u \neq v \in U$, let $x = \pi(u) \wedge_T \pi(v)$. Then $uv \in E(G)$ if and only if $(\kappa_{\mathfrak{T}}(u, x), \kappa_{\mathfrak{T}}(v, x)) \in \eta(x)$.

The *NLC-width* of a graph G is the minimum integer k such that there exists a k -NLC-tree that generates G (see Figure 2 for an example of k -NLC-tree).

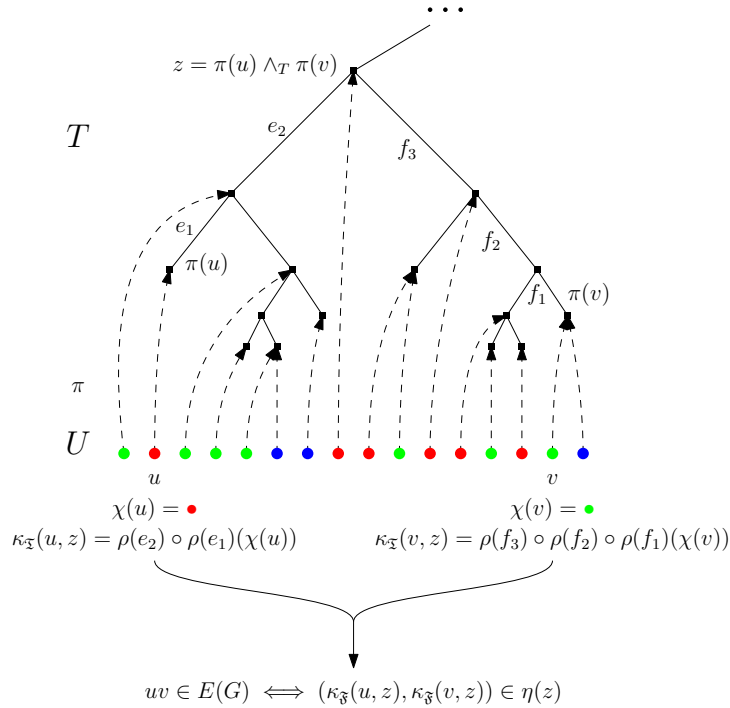


Figure 2: A k -NLC-tree $\mathfrak{T} = (T, U, \rho, \pi, \eta, \chi)$, and how the adjacency of two vertices u and v is determined.

Lemma 3.1 ([5, Lemma 3.6]). *For every integer k there exists a sequence of classes of k -NLC-trees $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \dots \subseteq \mathcal{F}_{3k^k}$ and a partition map $\mathfrak{T} \mapsto \mathcal{P}(\mathfrak{T})$, such that*

- (1) \mathcal{F}_0 contains only single node k -NLC-trees, while \mathcal{F}_{3k^k} is the class of all k -NLC-trees, and
- (2) for every $1 \leq i \leq 3k^k$ and every k -NLC-tree $\mathfrak{T} \in \mathcal{F}_i$, the factorization $\mathcal{P}(\mathfrak{T})$ of \mathfrak{T} is such that all the factors induced by parts of $\mathcal{P}(\mathfrak{T})$ belong to \mathcal{F}_{i-1} and the quotient tree $\mathfrak{T}/\mathcal{P}(\mathfrak{T})$ is either splendid or shallow.

Let \mathfrak{T} be a k -NLC-tree. The map $\mathcal{P}(\cdot)$ defines a *recursive factorization* of \mathfrak{T} , which can be represented as a rooted tree, whose root is \mathfrak{T} , where nodes are factors of \mathfrak{T} , and where the children of a factor \mathfrak{T}_F are the factors of \mathfrak{T}_F (thus of \mathfrak{T}) induced by the parts of $\mathcal{P}(\mathfrak{T}_F)$.

For a k -NLC-tree \mathfrak{T} , the *depth* of \mathfrak{T} is the minimum integer i such that $\mathfrak{T} \in \mathcal{F}_i$. Note that by [Lemma 3.1](#), the depth of \mathfrak{T} is always upper bounded by $3k^k$.

3.2 Proof of Theorem 1.2

In this section we prove that if a graph class \mathcal{C} has bounded rankwidth and stable edge relation, then \mathcal{C} can be transduced from a class of bounded treewidth. Therefore, let us fix positive integers k and h such that every graph in \mathcal{C} admits a k -NLC-tree and does not contain a half-graph of order h as a semi-induced subgraph.

We shall prove this inductively on the depth, as provided by [Lemma 3.1](#). More precisely, in the i th step of the induction we prove that graphs from \mathcal{C} that admit a k -NLC-tree belonging to \mathcal{F}_i can be transduced from a class of bounded treewidth. Since the depth of any k -NLC-tree is bounded by $3k^k$, the $3k^k$ th step of the induction will end the proof of [Theorem 1.2](#).

Therefore, let us fix some graph G and a k -NLC-tree $\mathfrak{T} = (T, U, \rho, \pi, \eta, \chi)$ generating G . We let $\mathcal{P} = \mathcal{P}(\mathfrak{T})$ be the factorization of \mathfrak{T} given by [Lemma 3.1](#), and we denote by (Y, U, ϱ, ϖ) the quotient \mathcal{S}_k -tree \mathfrak{T}/\mathcal{P} . Note that every factor of \mathcal{P} has depth lower than that of \mathfrak{T} , hence we may apply the induction assumption to it.

We first show how to handle the case when (Y, U, ϱ, ϖ) is splendid. Then we tackle the shallow case, which is significantly simpler. Each of these cases finishes with a technical claim summarizing the analysis. These claims are then used in a global induction scheme.

3.2.1 Splendid case

As (Y, U, ϱ, ϖ) is splendid, the set $R = \{\varrho(e) : e \in E(Y)\}$ is forward Ramsey. The following lemma shows that the recolorings then have a particularly nice form.

Lemma 3.2 (Claim 1 in Lemma 4.4 of [5]). *Let $R \subseteq \mathcal{S}_k$ be forward Ramsey. Then, for some $t \geq 1$, $[k]$ can be partitioned into parts $\gamma_1, \dots, \gamma_t$ so that for every $f \in R$ and every $i \in [t]$ there exists $m_i \in \gamma_i$ such that $f(m) = m_i$ for all $m \in \gamma_i$.*

By applying [Lemma 3.2](#) to R we obtain a suitable partition Γ of $[k]$. For $v \in U$, we let $\gamma(v)$ be the part of Γ that contains $\kappa_{\mathfrak{T}}(v, \text{top}(\varpi(v)))$. We call v a γ -vertex if $\gamma(v) = \gamma$.

Types and blocks. Throughout this section we use letters x, y, z etc. to denote the nodes of Y , which are parts of the factorization \mathcal{P} . For a node x of Y we denote by $P(x)$ the set of all ancestors of x in Y , except for x and its parent in Y , that is, $P(x) = \{y \in V(Y) : y \preceq_Y x^{\uparrow\uparrow}\}$. Recall that nodes of Y , being factors of T , are subtrees of T , hence it is meaningful to say that a node a of T belongs to a node x of Y .

Let x and y be two nodes of Y with $y \in P(x)$. Further, let $\gamma \in \Gamma$. Consider any vertex $v \in U$ satisfying $x \wedge_Y \varpi(v) = y$, and let $a = \text{top}(x) \wedge_T \pi(v)$. (Note that a is a vertex of y , considered as a subtree of T .) Then we say that x is (γ, y) -adjacent to v if for some (equivalently, every) $m \in \gamma$ we have

$$\left(\kappa_{\mathfrak{T}}(v, a), \rho(\text{path}_T(\text{top}(x), a))(m) \right) \in \eta(a).$$

Otherwise, we shall say that x is (γ, y) -non-adjacent to v . Note here that by the properties of Γ asserted by Lemma 3.2, the value of $\rho(\text{path}_T(\text{top}(x), a))(m)$ does not depend on the choice of $m \in \gamma$ whenever a does not belong to x or its parent in Y .

It may be useful to think of this definition as follows: if u and v are vertices in U , $x = \varpi(u)$, $y = x \wedge_Y \varpi(v)$, $y \in P(x)$, and $\kappa_{\mathbb{T}}(u, \text{top}(x)) \in \gamma$ (i.e. u is a γ -vertex), then u is adjacent to v in G if and only if x is (γ, y) -adjacent to v .

Fix $\gamma_0, \gamma_1 \in \Gamma$ and a node $x \in V(Y)$; possibly $\gamma_0 = \gamma_1$. For every node $y \in P(x)$, we define the (γ_0, γ_1) -type of y (seen from x), denoted $\text{tp}_{\gamma_0, \gamma_1}^x(y)$, as the pair $s_0 s_1$ where s_i is set as the symbol

- \circ if there is no γ_{1-i} -vertex $v \in U$ satisfying $x \wedge_Y \varpi(v) = y$; and otherwise:
- $+$ if x is (γ_i, y) -adjacent to all the γ_{1-i} -vertices $v \in U$ satisfying $x \wedge_Y \varpi(v) = y$;
- $-$ if x is (γ_i, y) -non-adjacent to all the γ_{1-i} -vertices $v \in U$ satisfying $x \wedge_Y \varpi(v) = y$; and
- \pm otherwise.

The following lemma proves a basic synchronization property: for two nodes $x_0, x_1 \in Y$, the types with respect to x_0 and x_1 synchronize above the parent of the least common ancestor of x_0 and x_1 .

Lemma 3.3. *If $x_0, x_1 \in Y$ and $y \in P(x_0 \wedge_Y x_1)$, then $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(y) = \text{tp}_{\gamma_0, \gamma_1}^{x_1}(y)$.*

Proof. Let $z = x_0 \wedge_Y x_1$. Consider any $m \in \gamma_0$. We have

$$\begin{aligned} \rho(\text{path}_Y(x_0, z^\uparrow))(m) &= \rho(\text{path}_Y(z, z^\uparrow)) \circ \rho(\text{path}_Y(x_0, z))(m) \\ &= \rho(\text{path}_Y(z, z^\uparrow)) \circ \rho(\text{path}_Y(x_1, z))(m) && \text{(by Lemma 3.2)} \\ &= \rho(\text{path}_Y(x_1, z^\uparrow))(m). \end{aligned}$$

Therefore, for every node $a \in V(T)$ that belongs to y and is an ancestor of $\text{top}(x_0)$ (equivalently, is an ancestor of $\text{top}(x_1)$), we have

$$\begin{aligned} \rho(\text{path}_T(\text{top}(x_0), a))(m) &= \rho(\text{top}(z^\uparrow), a) \circ \rho(\text{path}_Y(x_0, z^\uparrow))(m) \\ &= \rho(\text{top}(z^\uparrow), a) \circ \rho(\text{path}_Y(x_1, z^\uparrow))(m) \\ &= \rho(\text{path}_T(\text{top}(x_1), a))(m). \end{aligned}$$

It follows that the first coordinates of $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(y)$ and $\text{tp}_{\gamma_0, \gamma_1}^{x_1}(y)$ are equal. That the second coordinates are equal as well follows from a symmetric reasoning. \square

The next lemma contains the key combinatorial observation of the proof: a large alternation of types along $P(x)$ gives rise to a large half-graph as a semi-induced subgraph.

Lemma 3.4. *Suppose in $P(x)$ there are nodes*

$$z_\ell \prec_Y y_\ell \prec_Y z_{\ell-1} \prec_Y y_{\ell-1} \prec_Y \dots \prec_Y z_1 \prec_Y y_1$$

such that one of the following conditions holds:

- for each $i \in [\ell]$, the first coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(y_i)$ belongs to $\{+, \pm\}$ and the second coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(z_i)$ belongs to $\{-, \pm\}$;
- for each $i \in [\ell]$, the first coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(y_i)$ belongs to $\{-, \pm\}$ and the second coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(z_i)$ belongs to $\{+, \pm\}$.

Then $\ell \leq 3h$.

Proof. Let us assume that the first of the two conditions holds, as the proof in the second case is analogous. Suppose for contradiction that $\ell > 3h$. By the definition of types, for each $i \in [\ell]$ we can find a γ_1 -vertex v_i that is (γ_0, y_i) -adjacent to x . Similarly, for each $i \in [\ell]$ we can find a γ_0 -vertex w_i that is (γ_1, z_i) -non-adjacent to x . It easily follows from Lemma 3.3 (see Figure 4) that vertices

$$\{v_4, v_7, \dots, v_{3h+1}\} \quad \text{and} \quad \{w_2, w_5, w_8, \dots, w_{3h-1}\}$$

semi-induce a half-graph of order h in G , a contradiction. \square

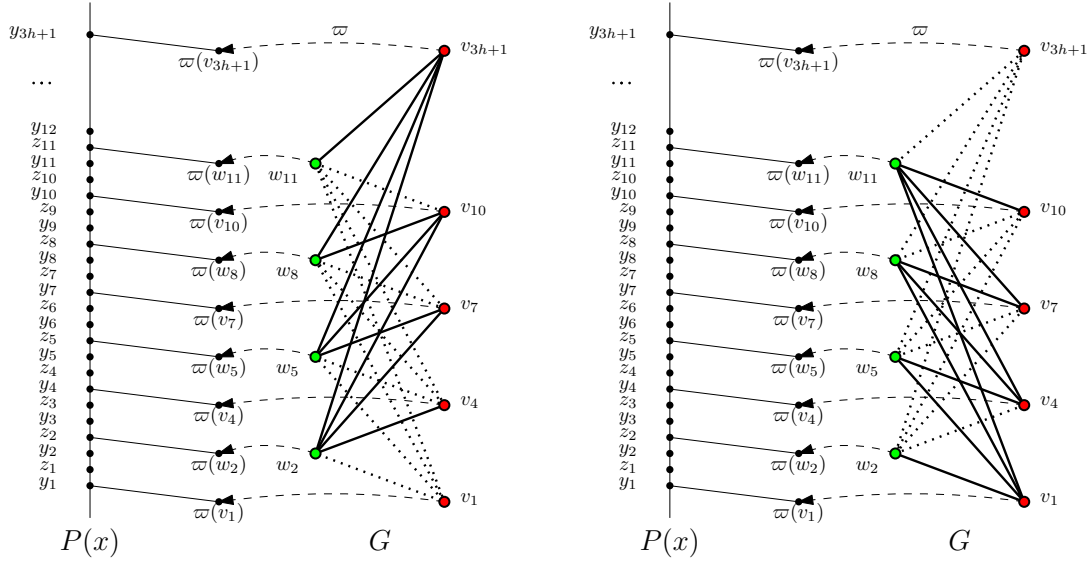


Figure 4: Illustration for Lemma 3.4. On the left, the first coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(y_i)$ belongs to $\{+, \pm\}$ and the second coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(z_i)$ belongs to $\{-, \pm\}$; on the right, the first coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(y_i)$ belongs to $\{-, \pm\}$ and the second coordinate of $\text{tp}_{\gamma_0, \gamma_1}^x(z_i)$ belongs to $\{+, \pm\}$. The green vertices are of class γ_0 , the red ones of class γ_1 . Thin edges correspond to paths in Y ; dashed arcs correspond to the ϖ mapping; fat edges correspond to edges (plain) and non edges (dotted) of G .

From Lemma 3.4 we may derive several structural properties of the sequence of types of nodes on $P(x)$. We consider $P(x)$ as a sequence ordered by the ancestor order, that is, the root of Y is the first element of this sequence. Let then $\text{Types}(x)$ be the sequence of types $\text{tp}_{\gamma_0, \gamma_1}^x(y)$ for $y \in P(x)$, ordered as in $P(x)$. In the following, by the *type* of $y \in P(x)$ we mean the type $\text{tp}_{\gamma_0, \gamma_1}^x(y)$.

We call an interval J in the sequence $\text{Types}(x)$ *valid* if it is of one of the following kinds:

- A *fully mixed* interval consists of a single node whose type does not contain \circ , but either contains \pm or both $+$ and $-$.
- A *positive* interval consists only of nodes with types in $\{\circ\circ, +\circ, \circ+, ++\}$.
- A *negative* interval consists only of nodes with types in $\{\circ\circ, -\circ, \circ-, --\}$.
- A *first-biased* interval consists only of nodes with types in $\{\circ\circ, -\circ, +\circ, \pm\circ\}$.
- A *second-biased* interval consists only of nodes with types in $\{\circ\circ, \circ-, \circ+, \circ\pm\}$.

Note that the cases are not exclusive. An interval that is either first- or second-biased will be just called *biased*. Note that a biased interval can be simultaneously positive and negative. If a first-biased (resp. a second-biased) interval J is neither positive nor negative (that is, it includes a symbol \pm or both symbols $+$ and $-$), then J is called *mixed-first-biased* (resp. *mixed-second-biased*).

For a node $y \in P(x)$, let $J(y)$ be the longest valid interval in $\text{Types}(x)$ that starts at the position corresponding to the node y . Then we define a partition $\text{Blocks}(x) = \{A_1, A_2, \dots\}$ of $\text{Types}(x)$ into subsequences, called *blocks*, via the following greedy procedure: if $P(x) = y_1 y_2 \dots y_\ell$, then

- $A_1 = J(y_1)$;
- $A_2 = J(y_{i_1+1})$, where y_{i_1} is the last element of A_1 ;
- $A_3 = J(y_{i_2+1})$, where y_{i_2} is the last element of A_2 , and so on.

The construction finishes once all the nodes of $P(x)$ are placed in the blocks. The blocks of $\text{Blocks}(x)$ are naturally ordered as in $\text{Types}(x)$, i.e. A_1 contains y_1 that is the root of Y .

The following lemma shows that the number of blocks in the sequence $\text{Blocks}(x)$ is always bounded in terms of h – the order of the half-graph that is forbidden in the graphs from \mathcal{C} . This is the key observation of the proof and, up to a technical reasoning, it follows from Lemma 3.4: many blocks give rise to a large half-graph in the generated graph.

Lemma 3.5. $\text{Blocks}(x)$ contains at most $60h + 9$ blocks.

Proof. For contradiction suppose $\text{Blocks}(x)$ contains more than $60h + 9$ blocks. Call a block *mixed* if it is either fully mixed, or mixed-first-biased, or mixed-second-biased.

Claim. $\text{Blocks}(x)$ contains at most $12h + 2$ fully-mixed blocks.

Proof of the claim. Suppose there are at least $12h + 3$ fully mixed blocks in $\text{Blocks}(x)$. Recall that each fully mixed block consists of a single node of type belonging to $\{+-, -+, \pm-, \pm+, -\pm, +\pm, \pm\pm\}$. Hence, we may either find at least $6k + 2$ nodes with types in the set $\{+-, +\pm, \pm-, \pm\pm\}$, or at least $6h + 2$ nodes with types in the set $\{-+, -\pm, \pm+, \pm\pm\}$. In both cases, these at least $6h + 2$ nodes form a structure that is forbidden by Lemma 3.4, a contradiction. \triangleleft

Claim. $\text{Blocks}(x)$ contains at most $6h + 1$ mixed-first-biased blocks and at most $6h + 1$ mixed-second-biased blocks.

Proof of the claim. We prove the bound on the number of mixed-first-biased blocks. The bound for mixed-second-biased blocks follows analogously with the roles of γ_0 and γ_1 exchanged.

Suppose for contradiction that there are more than $6h + 1$ mixed-first-biased blocks in $\text{Blocks}(x)$. Let X_1, \dots, X_{6h+2} be any $6h + 2$ of them, ordered as in $\text{Types}(x)$. For $i \in [6h + 1]$, let z_i be the node of $P(x)$ that immediately follows the last node of X_i . Note that by the construction of $\text{Blocks}(x)$, the second coordinate of the type of z_i cannot be \circ , for otherwise z_i would be in X_i . In particular, $z_i \notin X_{i+1}$ and z_i lies in $P(x)$ strictly before X_{i+1} .

As argued, for each $i \in [6h + 1]$ the second coordinate of the type of z_i belongs to $\{-, +, \pm\}$. Therefore, there exists a subset of indices $I \subseteq [6h + 1]$ of size $3h + 1$ such that either for each $i \in I$, the second coordinate of the type of z_i belongs to $\{-, \pm\}$, or for each $i \in I$, the second coordinate of the type of z_i belongs to $\{+, \pm\}$. Assume the former case, as the proof in the latter case is symmetric.

Since each X_i is a mixed-first-biased block, for each $i \in I$ we may find a node $y_i \in X_i$ such that the first coordinate of the type of y_i belongs to $\{+, \pm\}$. Now the nodes $\{y_i, z_i : i \in I\}$ form a structure forbidden by Lemma 3.4, a contradiction. \triangleleft

By the above claims, the total number of mixed blocks is at most $24k + 4$. Call a block *unaffected* if it is not mixed and the block succeeding it exists and is not mixed either. Then the total number of unaffected blocks is larger than $(60h + 9) - 2 \cdot (24h + 4) - 1 = 12h$. Out of these, there are either more than $6h$ unaffected positive blocks, or more than $6h$ unaffected negative blocks. Assume the former case, as the proof in the latter case is symmetric.

Let then B_1, \dots, B_{6h+1} be any $6h + 1$ unaffected positive blocks, and let C_1, \dots, C_{6h+1} be the successors of blocks B_1, \dots, B_{6h+1} , respectively. Since B_1, \dots, B_{6h+1} are unaffected and positive, it follows that C_1, \dots, C_{6h+1} are negative blocks. Observe that for each $i \in [6h + 1]$, it cannot happen that for all the nodes $t \in B_i \cup C_i$, the first coordinate of the type of t is \circ . Indeed, then $B_i \cup C_i$ would be a first-biased interval, and therefore it would be a valid interval that would contain the block B_i as a prefix. Similarly, for each $i \in [6h + 1]$, it cannot happen that the second coordinate of the type of t is \circ for all $t \in B_i \cup C_i$. We conclude that for each $i \in [6h + 1]$, we may find nodes $y_i \in B_i$ and $z_i \in C_i$ such that one of the following alternatives holds:

- the first coordinate of the type of y_i is not \circ (and therefore must be $+$) and the second coordinate of the type of z_i is not \circ (and therefore must be $-$); or
- the second coordinate of the type of y_i is not \circ (and therefore must be $+$) and the first coordinate of the type of z_i is not \circ (and therefore must be $-$).

By the pigeonhole principle, one of these two alternatives holds for at least $3h + 1$ indices $i \in [6h + 1]$. Suppose this is the first alternative, as the proof in the other case proceeds analogously with the roles of γ_0 and γ_1 exchanged. It now follows that if $I \subseteq [6h + 1]$ is a set of size $3h + 1$ such that the first alternative holds for each $i \in I$, then the nodes $\{y_i, z_i : i \in I\}$ form a structure forbidden by Lemma 3.4, a contradiction. \square

For a node x of Y , we define the following:

- $Q(x)$ is the set consisting of x and the parent of x in Y , if existent;
- $S(x)$ is the set containing, for each block $A \in \text{Blocks}(x)$, the \preceq_Y -minimal element of A , the \preceq_Y -minimal element of A whose type belongs to $\{-, \pm\}$ (if existent), and the \preceq_Y -minimal element of A whose type belongs to $\{+, \pm\}$ (if existent);
- for each $\gamma \in \Gamma$, $g_\gamma(x)$ is the \preceq_Y -maximal ancestor of x such that there exists a γ -vertex w satisfying $g_\gamma(x) \preceq_Y \varpi(w)$, or $g_\gamma(x) = \perp$ if no such ancestor exists.

Further, let

$$\mathbb{L}(x) = Q(x) \cup S(x).$$

By Lemma 3.5, we have

$$|\mathbb{L}(x)| \leq 2 + 3 \cdot (60h + 9) \leq 209h.$$

Intuitively, $\mathbb{L}(x) \cup \{g_\gamma(x) : \gamma \in \Gamma\}$ contains all vertices that are interesting from the point of view of x .

Recovering edges: combinatorial analysis. Let us fix two vertices $u_0, u_1 \in U$. Let

$$\begin{aligned} x_0 &= \varpi(u_0), & x_1 &= \varpi(u_1), \\ \gamma_0 &= \gamma(u_0), & \gamma_1 &= \gamma(u_1). \end{aligned}$$

Adopting the notation from the previous section, we have sets $P(x_0)$ and $P(x_1)$ and their partitions $\text{Blocks}(x_0)$ and $\text{Blocks}(x_1)$. Intuitively, our goal is to show that given sets $\mathbb{L}(x_0)$ and $\mathbb{L}(x_1)$, we may either directly infer whether u_0 and u_1 are adjacent in G , or locate the node $z = x_0 \wedge_Y x_1$, that is, the lowest common ancestor of x_0 and x_1 . In the subsequent section we will implement this mechanism in first-order logic. Lemma 3.3 implies that the sequences of types $\text{Types}(x_0)$ and $\text{Types}(x_1)$ agree on the prefix up to the grandparent of z .

Let Z be the set consisting of:

- z ;
- the parent of z , if existent;
- the child of z that is an ancestor of x_0 , if existent; and
- the child of z that is an ancestor of x_1 , if existent.

We will further work under the following assumption:

$$\mathbb{L}(x_0) \cap Z = \emptyset \quad \text{or} \quad \mathbb{L}(x_1) \cap Z = \emptyset. \quad (*)$$

Intuitively, if assumption $(*)$ is not satisfied, then both $\mathbb{L}(x_0)$ and $\mathbb{L}(x_1)$ contain either z or its neighbor in Y , and then locating z will be easy.

Note that the root of Y always belongs to $\mathbb{L}(x_0) \cap \mathbb{L}(x_1)$. Hence, assuming $(*)$, z is neither the root of Y nor a child of the root of Y . Then both $P(x_0)$ and $P(x_1)$ are non-empty, implying that also $\text{Blocks}(x_0)$ and $\text{Blocks}(x_1)$ are non-empty. Let

$$\text{Blocks}(x_0) = \{A_1, A_2, \dots, A_p\} \quad \text{and} \quad \text{Blocks}(x_1) = \{B_1, B_2, \dots, B_q\},$$

where blocks A_i and B_j are ordered naturally by the ancestor order so that the root of Y belongs to A_1 and B_1 . For a block A_i , let $\text{top}(A_i)$ be the first (i.e. \preceq_Y -minimal) node of A_i ; define $\text{top}(B_j)$ analogously.

Let i be the largest index such that $\text{top}(A_i) = \text{top}(B_i)$. Note that i is well-defined, because $\text{top}(A_1) = \text{top}(B_1)$. Let $t = \text{top}(A_i) = \text{top}(B_i)$. Since t is both an ancestor of x_0 and of x_1 , we have $t \preceq_Y z$. Furthermore, since $t \in \mathbb{L}(x_0) \cap \mathbb{L}(x_1)$, from $(*)$ we infer that $t \notin Z$.

Lemma 3.6. *The node z has the following properties:*

1. $z \in Q(x_0)$ or the first coordinate of $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(z)$ is not equal to \circ ;

2. $z \in Q(x_1)$ or the second coordinate of $\text{tp}_{\gamma_0, \gamma_1}^{x_1}(z)$ is not equal to \circ ;
3. $z \in A_i \cup B_i$.

Proof. The first two points follow directly from the existence of vertices u_0 and u_1 . We are left with arguing that $z \in A_i \cup B_i$. Suppose otherwise. Then both A_{i+1} and B_{i+1} exist, and moreover $\text{top}(A_{i+1}) \prec_Y z$ and $\text{top}(B_{i+1}) \prec_Y z$. By the maximality of i we have $\text{top}(A_{i+1}) \neq \text{top}(B_{i+1})$.

By [Lemma 3.3](#) and the construction of $\text{Blocks}(x_1)$ and $\text{Blocks}(x_2)$, every ancestor of the grandparent of z is the top vertex of a block in $\text{Blocks}(x_1)$ if and only if it is the top vertex of a block in $\text{Blocks}(x_2)$. Therefore, $\text{top}(A_{i+1}) \neq \text{top}(B_{i+1})$ together with $\text{top}(A_{i+1}) \prec_Y z$ and $\text{top}(B_{i+1}) \prec_Y z$ implies that $\text{top}(A_{i+1}) \in Z$ and $\text{top}(B_{i+1}) \in Z$. As $\text{top}(A_{i+1}) \in \mathbb{L}(x_0)$ and $\text{top}(B_{i+1}) \in \mathbb{L}(x_1)$, this contradicts assumption (*). \square

Let $R := \{r : t \preceq_Y r \preceq_Y z \text{ and } r \notin Z\}$. Note that $t \in R$, hence R is non-empty. By [Lemma 3.3](#), we have

$$\text{tp}_{\gamma_0, \gamma_1}^{x_0}(r) = \text{tp}_{\gamma_0, \gamma_1}^{x_1}(r) \quad \text{for each } r \in R. \quad (1)$$

From the construction of $\text{Blocks}(x_0)$ and of $\text{Blocks}(x_1)$ it then follows that

$$R \subseteq A_i \cap B_i. \quad (2)$$

We now observe the following.

Lemma 3.7. *There exists $r \in R$ such that*

$$\text{tp}_{\gamma_0, \gamma_1}^{x_0}(r) = \text{tp}_{\gamma_0, \gamma_1}^{x_1}(r) \neq \circ\circ.$$

Proof. Suppose otherwise: $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(r) = \text{tp}_{\gamma_0, \gamma_1}^{x_1}(r) = \circ\circ$ for all $r \in R$. By [Lemma 3.6](#), we either have $z \in Q(x_0)$, or $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(z) \neq \circ\circ$. The latter condition implies that either A_{i+1} exists and $\text{top}(A_{i+1}) \in Z$, or the \preceq_Y -minimal element of block A_i whose type features a non- \circ symbol belongs to Z . In each of these three cases we have $\mathbb{L}(x_0) \cap Z \neq \emptyset$. A symmetric reasoning shows that also $\mathbb{L}(x_1) \cap Z \neq \emptyset$. This is a contradiction with assumption (*). \square

We introduce the following notation. For $\gamma \in \Gamma$ and $y \in V(Y)$, if there is a unique grandchild y' of y in Y such that for every γ -vertex v satisfying $y \preceq_Y \varpi(v)$ we have $y' \preceq_Y \varpi(v)$, then we set $h_\gamma(y) = y'$. If there is no such grandchild, we set $h_\gamma(y) = \perp$.

Lemma 3.8. *None of the blocks A_i or B_i is fully mixed. Moreover, depending on the kinds the blocks A_i and B_i belong to, we have the following cases:*

1. *If A_i is not biased, then*
 - *either A_i is positive and $u_0 u_1 \in E(G)$,*
 - *or A_i is negative and $u_0 u_1 \notin E(G)$.*
2. *If B_i is not biased, then*
 - *either B_i is positive and $u_0 u_1 \in E(G)$,*
 - *or B_i is negative and $u_0 u_1 \notin E(G)$.*
3. *If both A_i and B_i are biased, then*
 - *either both A_i and B_i are first-biased, and then $h_{\gamma_0}(z) \neq \perp$,*
 - *or both A_i and B_i are second-biased and then $h_{\gamma_1}(z) \neq \perp$.*

Proof. First, we observe the following.

Claim. None of the blocks A_i or B_i is fully mixed.

Proof of the claim. Recall that a fully mixed block consists of one node whose type does not feature symbol \circ , but features either \pm or both $+$ and $-$. Therefore, if any of A_i or B_i was fully mixed, then both of them would be, implying that $A_i = B_i = \{t\}$. This stands in contradiction with Lemma 3.6. \triangleleft

Next, we treat the case when A_i or B_i is not biased.

Claim. Suppose A_i is not biased. Then exactly one of the following holds: A_i is positive and $u_0u_1 \in E(G)$, or A_i is negative and $u_0u_1 \notin E(G)$. Symmetrically, supposing B_i is not biased, exactly one of the following holds: B_i is positive and $u_0u_1 \in E(G)$, or B_i is negative and $u_0u_1 \notin E(G)$.

Proof of the claim. We prove the first assertion; the reasoning proving the second one is symmetric.

By the previous claim and the assumption, A_i is neither fully mixed, nor first-biased, nor second-biased. Therefore, A_i is either positive or negative. Note that by Lemma 3.7 and (2), A_i cannot be both positive and negative at the same time. It remains to prove that if A_i is positive, then $u_0u_1 \in E(G)$; the proof that A_i being negative entails $u_0u_1 \notin E(G)$ is symmetric.

Note that if we have $z \in A_i$, then A_i being positive immediately implies that $u_0u_1 \in E(G)$. Therefore, suppose that $z \notin A_i$, which implies that $z \preceq \text{top}(A_{i+1})$ and as $\text{top}(B_{i+1}) \neq \text{top}(A_{i+1})$ (by definition of i), $z \in \mathbb{L}(x_0)$ and thus $\mathbb{L}(x_0) \cap Z \neq \emptyset$. By Lemma 3.6, we have $z \in B_i$. Suppose for contradiction that $u_0u_1 \notin E(G)$. Then the second coordinate of $\text{tp}_{\gamma_0, \gamma_1}^{x_1}(z)$ has to be either $-$ or \pm . However, since A_i is positive, from (1) and (2) we infer that types $\text{tp}_{\gamma_0, \gamma_1}^{x_1}(r)$ for $r \in R$ feature only symbols \circ and $+$. Therefore, the \preceq_Y -minimal element of B_i that contains symbol $-$ or \pm is either z or its parent, implying that $\mathbb{L}(x_1) \cap Z \neq \emptyset$. Together with $\mathbb{L}(x_0) \cap Z \neq \emptyset$, this contradicts assumption (*). \triangleleft

We are left with the case when both A_i and B_i are biased. First, we observe that they need to be biased in the same direction.

Claim. If both A_i and B_i are biased, then exactly one of the following holds: both A_i and B_i are first-biased, or both A_i and B_i are second-biased.

Proof of the claim. Follows directly from Lemma 3.7 together with (2). \triangleleft

We now show how to locate z in this case.

Claim. Suppose A_i and B_i are both first-biased. Then $z \in A_i$ and $z = g_{\gamma_0}(x_1)$; in particular $z \in P(x_0)$. Moreover, there exists a grandchild z_0 of z such that for every γ_0 -vertex v satisfying $z \preceq_Y \varpi(v)$, we in fact have $z_0 \preceq_Y \varpi(v)$. Also, there exist γ_0 -vertices satisfying this condition.

In other words, $h_{\gamma_0}(z) = z_0$.

Proof of the claim. Since B_i is first-biased, from Lemma 3.6(2) we infer that $z \in Q(x_1)$. It implies that $\mathbb{L}(x_1) \cap Z \neq \emptyset$ and that $z \notin P(x_1)$ thus $z \notin B_i$. By Lemma 3.6(3), $z \in A_i$. As $z \notin B_i$ and $R \subseteq B_i$, we have $\mathbb{L}(x_1) \cap Z \neq \emptyset$. Therefore, from assumption (*) we conclude that $\mathbb{L}(x_0) \cap Z = \emptyset$.

As $z \in A_i$, we in particular have $z \in P(x_0)$, hence z is neither x_0 nor the parent of x_0 . Let then z_0 be the grandchild of z such that $z_0 \preceq_Y x_0$. Further, let z'_0 be the parent of z_0 . Note that $z'_0 \in Z$. Since $\mathbb{L}(x_0) \cap Z = \emptyset$, we must have $z'_0 \in A_i$.

Since A_i is first-biased and $z, z'_0 \in A_i$, the second coordinates of $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(z)$ and of $\text{tp}_{\gamma_0, \gamma_1}^{x_0}(z'_0)$ are both \circ . Therefore, there are no γ_0 -vertices v satisfying $x_0 \wedge_Y \varpi(v) = z$ or $x_0 \wedge_Y \varpi(v) = z'_0$, which means that for every γ_0 -vertex v satisfying $z \preceq_Y \varpi(v)$, we in fact have $z_0 \preceq_Y \varpi(v)$. That there exist γ_0 -vertices satisfying this condition is witnessed by u_0 . \triangleleft

A symmetric reasoning yields the following.

Claim. Suppose A_i and B_i are both second-biased. Then $z \in B_i$ and $z = g_{\gamma_1}(x_0)$; in particular $z \in P(x_1)$. Moreover, there exists a grandchild z_1 of z such that for every γ_1 -vertex v satisfying $z \preceq_Y \varpi(v)$, we in fact have $z_1 \preceq_Y \varpi(v)$. Also, there exist γ_1 -vertices satisfying this condition.

In other words, $h_{\gamma_1}(z) = z_1$.

The presented claims verify all the assertions from the lemma statement. \square

Recovering edges: logical implementation. We now define a structure $H_{\mathfrak{T}}$ which encodes all the relevant information about the k -NLC-tree \mathfrak{T} and its factorization \mathcal{P} . Intuitively, $H_{\mathfrak{T}}$ encodes \mathfrak{T} in the natural way, plus in addition we enrich it with pointers encoding sets $\mathbb{L}(x)$ and functions $g_\gamma(x), h_\gamma(x)$.

Formally, the universe of $H_{\mathfrak{T}}$ is just $V(T)$; note that the set U will *not* be directly encoded. In $H_{\mathfrak{T}}$ we will use only unary predicates and unary (partial) functions. Of course, the latter can be replaced by suitable functional binary relations in order to make the signature purely relational. In the following, whenever we encode some node y that belongs to the quotient tree Y , we represent it using $\text{top}(y)$. For instance, the parent function in Y is represented as a partial function on the nodes of T that maps $\text{top}(x)$ to $\text{top}(x')$ whenever x' is the parent of x in Y .

For $x \in V(Y)$, let $\widehat{\mathbb{L}}(x) \subseteq V(Y)$ be the set containing every ancestor of x that:

- belongs to $\mathbb{L}(x)$,
- is the parent of a node of $\mathbb{L}(x)$,
- is the child of a node of $\mathbb{L}(x)$ on $P(x)$, or
- is the grandchild of a node of $\mathbb{L}(x)$ on $P(x)$.

Recalling that $|\mathbb{L}(x)| \leq 209h$, we have $|\widehat{\mathbb{L}}(x)| \leq 836h$. Also, for $x \in V(Y)$ and $\gamma \in \Gamma$, we let $\widehat{g}_\gamma(x)$ be the child of $g_\gamma(x)$ that is an ancestor of x . In case $g_\gamma(x) = x$, we set $\widehat{g}_\gamma(x) = \perp$.

In the following encoding, all values featuring \perp are removed from the domains of corresponding mappings. Then, in $H_{\mathfrak{T}}$ we encode:

- the parent function of the tree T ;
- the parent function of the tree Y ;
- the mapping $a \mapsto \rho(e(a))$, where a is a node of T and $e(a)$ is the edge of T connecting a with its parent;
- the mapping $x \mapsto \varrho(e(x))$, where x is a node of Y and $e(x)$ is the edge of Y connecting x with its parent;
- the mappings $a \mapsto \text{top}(x(a))$ and $a \mapsto \rho(\text{path}_T(a, \text{top}(x(a))))$, where a is a node of T and $x(a)$ is the node of Y such that $a \in x(a)$;
- for each $\gamma \in \Gamma$, the mappings $x \mapsto g_\gamma(x)$, $x \mapsto \widehat{g}_\gamma(x)$, and $x \mapsto h_\gamma(x)$;
- for each $\gamma \in \Gamma$, the mapping $x \mapsto \varrho(\text{path}_Y(x, \widehat{g}_\gamma(x)))$;
- the mapping $x \mapsto \widehat{\mathbb{L}}(x)$, together with relevant data about the elements of $\widehat{\mathbb{L}}(x)$; and
- for every node x of Y and $y \in \widehat{\mathbb{L}}(x)$, the value $\varrho(\text{path}_Y(x, y))$.

Here, the last two points require more explanation. Recall that $|\widehat{\mathbb{L}}(x)| \leq 836h$ for each $x \in V(Y)$. Therefore, to encode the mapping $x \mapsto \widehat{\mathbb{L}}(x)$ we use $836h$ distinct unary functions, where the i th function maps a node $x \in V(Y)$ to the i th element of $\widehat{\mathbb{L}}(x)$, sorted by the ancestor order. The relevant data about a node $y \in \widehat{\mathbb{L}}(x)$ includes whether y is the \preceq_Y -minimal node of some block of $\text{Blocks}(x)$ and if so, what kind of block it is (positive or negative, first-biased or second-biased, etc.). This information can be encoded using unary predicates at x . Similarly, to encode the values $\varrho(\text{path}_Y(x, y))$ for $y \in \widehat{\mathbb{L}}(x)$, we use $836h$ distinct unary predicates at x , where the i th predicate encodes $\varrho(\text{path}_Y(x, y))$ where y is the i th element of $\widehat{\mathbb{L}}(x)$.

We later use some properties of $H_{\mathfrak{T}}$ that follow from the synchronization property expressed by [Lemma 3.3](#). For this, for a node a of T , we define $N^\uparrow(a)$ to be the set of all nodes b of T such that $b \prec_T a$ and there is a function f in $H_{\mathfrak{T}}$ such that $b = f(a)$ or $a = f(b)$. Then we have the following.

Lemma 3.9. For each $a \in V(T)$,

$$\left| \{b \in V(T) : b \prec_T a\} \cap \bigcup_{a' \succeq_T a} N^\uparrow(a') \right| \leq 836h + 2k + 4.$$

Proof. Let $x \in V(Y)$ be such that $a \in x$. From Lemma 3.3 and the construction of the blocks it follows that for all $x', x'' \in V(Y)$ such that $\text{top}(x'), \text{top}(x'') \succeq_T a$, we have

$$\mathbb{L}(x') \cap P(x) = \mathbb{L}(x'') \cap P(x).$$

Thus,

$$\widehat{\mathbb{L}}(x') \cap P(x^\uparrow) = \widehat{\mathbb{L}}(x'') \cap P(x^\uparrow)$$

for all such x', x'' . Let M_0 be this common subset of $P(x^\uparrow)$; note that $\widehat{\mathbb{L}}(x') \cap P(x) \subseteq M_0 \cup \{x^\uparrow\}$. Let $M'_0 = \{\text{top}(z) : z \in M_0\} \cup \{\text{top}(x^\uparrow)\}$; then $|M'_0| \leq 836h + 1$.

Similarly, for all x', x'' as above, we have

$$\{g_\gamma(x'), \widehat{g}_\gamma(x') : \gamma \in \Gamma\} \cap P(x) = \{g_\gamma(x''), \widehat{g}_\gamma(x'') : \gamma \in \Gamma\} \cap P(x),$$

so let M_1 be this common subset of $P(x)$ and let $M'_1 = \{\text{top}(z) : z \in M_1\}$. Note that $|M'_1| \leq 2|\Gamma| \leq 2k$. It can now be easily seen from the construction of $H_{\overline{\mathfrak{T}}}$ that for each $a' \succeq_T a$, we have

$$\{b \in V(T) : b \preceq_T a'\} \cap N^\uparrow(a') \subseteq M'_0 \cup M'_1 \cup \{a^\uparrow, \text{top}(x), \text{top}(x^\uparrow)\}.$$

Since the set on the right hand side has size at most $836h + 2k + 4$, the claim follows. \square

Our next goal is to implement the combinatorial analysis described in the previous section using first-order formulas working over $H_{\overline{\mathfrak{T}}}$. Before we do this, let us see how the information about elements of U can be recovered from $H_{\overline{\mathfrak{T}}}$. Suppose $u \in U$ is a vertex for which we know that $\pi(u) = a$ and $\chi(u) = c$. Then $\varpi(u)$ can be easily inferred as $\text{top}(x(a))$. Similarly, the color $\kappa_{\overline{\mathfrak{T}}}(u, \varpi(u))$ can be obtained by applying $\rho(\text{path}_T(a, \text{top}(x(a))))$ to c . This in particular gives the value of $\gamma(u)$. Finally, whenever for some ancestor y of $x = \varpi(u)$, the value of $\rho(\text{path}_Y(x, y))$ is stored in $H_{\overline{\mathfrak{T}}}$, then the color $\kappa_{\overline{\mathfrak{T}}}(u, \text{top}(y))$ can be obtained by applying $\rho(\text{path}_Y(x, y))$ to $\kappa_{\overline{\mathfrak{T}}}(u, \varpi(u))$. This may happen when $y = \widehat{g}_\gamma(x)$ for some $\gamma \in \Gamma$, or when $y \in \widehat{\mathbb{L}}(x)$.

We are now ready to provide the promised implementation.

Lemma 3.10. *Fix $c_0, c_1 \in [k]$. Then there are formulas*

$$\varphi_{c_0, c_1}(p_0, p_1), \quad \psi_{c_0, c_1}(p_0, p_1), \quad \text{and} \quad \{\zeta_{c_0, c_1, d_0, d_1}(p_0, p_1, q, q_0, q_1) : d_0, d_1 \in [k]\}$$

in the vocabulary of $H_{\overline{\mathfrak{T}}}$ such that the following holds for all distinct $u_0, u_1 \in U$ satisfying $\chi(u_0) = c_0$ and $\chi(u_1) = c_1$, where $a_0 = \pi(u_0)$ and $a_1 = \pi(u_1)$.

- *If $H_{\overline{\mathfrak{T}}} \models \varphi_{c_0, c_1}(a_0, a_1)$, then u_0 and u_1 are adjacent in G if and only if $H_{\overline{\mathfrak{T}}} \models \psi_{c_0, c_1}(a_0, a_1)$.*
- *If $H_{\overline{\mathfrak{T}}} \not\models \varphi_{c_0, c_1}(a_0, a_1)$, then there is a unique 5-tuple $(d_0, d_1, t, t_0, t_1) \in [k]^2 \times V(T)^3$ such that $H_{\overline{\mathfrak{T}}} \models \zeta_{c_0, c_1, d_0, d_1}(a_0, a_1, t, t_0, t_1)$:*
 - $t = \text{top}(\varpi(u_0) \wedge_Y \varpi(u_1))$;
 - t_0 is the \preceq_T -maximum node of $\varpi(u_0) \wedge_Y \varpi(u_1)$ satisfying $t_0 \preceq_T \pi(u_0)$;
 - t_1 is the \preceq_T -maximum node of $\varpi(u_0) \wedge_Y \varpi(u_1)$ satisfying $t_1 \preceq_T \pi(u_1)$;
 - $d_0 = \kappa_{\overline{\mathfrak{T}}}(u_0, t_0)$; and
 - $d_1 = \kappa_{\overline{\mathfrak{T}}}(u_1, t_1)$.

Proof. We explain how, given $a_0, a_1 \in V(T)$, $c_0, c_1 \in [k]$, and access to the information present in $H_{\overline{\mathfrak{T}}}$, to either determine whether u_0 and u_1 are adjacent in G or not, or find the 5-tuple (d_0, d_1, t, t_0, t_1) described in the statement. It is straightforward to encode the explained mechanism in first-order logic, which gives rise to the postulated first-order formulas.

Let us adopt the notation from the previous section for u_0 and u_1 . In particular, u_0 is a γ_0 -vertex, u_1 is a γ_1 -vertex, $\varpi(u_0) = x_0$, $\varpi(u_1) = x_1$, and $z = x_0 \wedge_Y x_1$. As argued, $\gamma_0, \gamma_1, x_0, x_1$ can be inferred from c_0, c_1, a_0, a_1 given access to $H_{\overline{\mathfrak{T}}}$.

As the first step, we find the \preceq_Y -maximal element of $\widehat{\mathbb{L}}(x_0) \cap \widehat{\mathbb{L}}(x_1)$. Call it \tilde{z} . First, we consider the corner case when $x_0 = x_1 = \tilde{z}$. Then we have:

- $t = \text{top}(x_0) = \text{top}(x_1)$;
- $t_0 = a_0$;
- $t_1 = a_1$;
- $d_0 = c_0$; and
- $d_1 = c_1$.

Second, we check whether both $\widehat{\mathbb{L}}(x_0)$ and $\widehat{\mathbb{L}}(x_1)$ contain a child of \tilde{z} . Suppose for a moment that this is the case, and let z'_0 and z'_1 be these children, respectively. Then by the maximality of \tilde{z} , we must have $z'_0 \neq z'_1$, implying $z = \tilde{z}$. It follows that:

- $t = \text{top}(z)$;
- t_0 is the parent in T of $\text{top}(z'_0)$;
- t_1 is the parent in T of $\text{top}(z'_1)$;
- $d_0 = \rho(e(\text{top}(z'_0))) (\kappa_{\mathfrak{T}}(u_0, \text{top}(z'_0)))$; and
- $d_1 = \rho(e(\text{top}(z'_1))) (\kappa_{\mathfrak{T}}(u_1, \text{top}(z'_1)))$.

As we argued, these values can be retrieved from $H_{\mathfrak{T}}$ given c_0, c_1, a_0, a_1 .

Next, we consider a mix of the two cases above: $x_0 = \tilde{z}$ and \tilde{z} has a child z'_1 that belongs to $\widehat{\mathbb{L}}(x_1)$. Then again we have $z = \tilde{z}$ and:

- $t = \text{top}(z)$;
- $t_0 = a_0$;
- t_1 is the parent in T of $\text{top}(z'_1)$;
- $d_0 = c_0$; and
- $d_1 = \rho(e(\text{top}(z'_1))) (\kappa_{\mathfrak{T}}(u_1, \text{top}(z'_1)))$.

The case when $x_1 = \tilde{z}$ and \tilde{z} has a child z'_0 that belongs to $\widehat{\mathbb{L}}(x_0)$ is symmetric.

We claim that the four cases considered above cover all the situations when assumption $(*)$ is not satisfied, that is, when $\mathbb{L}(x_0) \cap Z \neq \emptyset$ and $\mathbb{L}(x_1) \cap Z \neq \emptyset$. Indeed, if this is the case, then $\widehat{\mathbb{L}}(x_0)$ and $\widehat{\mathbb{L}}(x_1)$ both contain z . Moreover, $\widehat{\mathbb{L}}(x_0)$ contains the child of z that is an ancestor of x_0 , if existent, and similarly $\widehat{\mathbb{L}}(x_1)$ contains the child of z that is an ancestor of x_1 , if existent. Then $z = \tilde{z}$ and in either way, one of the four cases considered above applies.

Hence, from now on we proceed under the assumption that $(*)$ holds. Consequently, all the claims presented in the previous section can be applied.

Denoting $\mathcal{P}(x_0) = \{A_1, \dots, A_p\}$ and $\mathcal{P}(x_1) = \{B_1, \dots, B_q\}$, we find the largest index i such that $\text{top}(A_i) = \text{top}(B_i)$. Note that i and the kinds to which blocks A_i and B_i belong can be retrieved using the information stored along with sets $\widehat{\mathbb{L}}(x_0)$ and $\widehat{\mathbb{L}}(x_1)$.

By Lemma 3.8, none of the blocks A_i or B_i can be fully mixed. If either A_i or B_i is not biased, we may use Lemma 3.8-(1) and Lemma 3.8-(2) to directly infer whether u_0 and u_1 are adjacent in G or not. We are left with the case when both A_i and B_i are biased. By Lemma 3.8-(3), they are either both first-biased, or both second-biased.

Suppose that both A_i and B_i are first-biased. Then, by Lemma 3.8, we have:

- $z = g_{\gamma_0}(x_1)$;
- $z_0 = h_{\gamma_0}(z) \neq \perp$;
- if z'_0 is the parent in Y of z_0 , then t_0 is the parent in T of $\text{top}(z'_0)$; and
- if d'_0 is the unique element of $\varrho(e(z_0))(\gamma_0)$, then $d_0 = \rho(e(\text{top}(z'_0)))(d'_0)$.

Here, the fact that $\varrho(e(z_0))(\gamma_0)$ consists of exactly one element of γ_0 is implied by the fact that \mathfrak{T}/\mathcal{P} is splendid, as asserted by Lemma 3.2. It remains to retrieve t_1 and d_1 . For this, by Lemma 3.8 we observe that if $\widehat{g}_{\gamma_0}(x_1) = \perp$ then $x_1 = z$ and we have

- $t_1 = a_1$ and
- $d_1 = c_1$.

Otherwise, if $\widehat{g}_{\gamma_0}(x_1) \neq \perp$, then $\widehat{g}_{\gamma_0}(x_1)$ is the ancestor of x_1 that is a child of z and we have:

- t_1 is the parent in T of $\text{top}(\widehat{g}_{\gamma_0}(x_1))$ and
- $d_1 = \rho(e(\widehat{g}_{\gamma_0}(x_1))) (\kappa_{\mathfrak{T}}(u_1, \text{top}(\widehat{g}_{\gamma_0}(x_1))))$.

The case when both A_i and B_i are second-biased is symmetric. As in all the cases we have either concluded whether u_0 and u_1 are adjacent or not, or we have determined the 5-tuple (d_0, d_1, t, t_0, t_1) , this finishes the proof. \square

3.2.2 Shallow case

We now treat the case when the quotient tree (Y, U, ϱ, ϖ) is shallow; recall that this means that Y has height 1. As in the previous section, we encode \mathfrak{T} in a structure $H_{\mathfrak{T}}$ whose universe is $V(T)$. We encode the following information in $H_{\mathfrak{T}}$:

- the parent function of the tree T ;
- the mapping $a \mapsto \rho(e(a))$, where a is a node of T and $e(a)$ is the edge of T connecting a with its parent;
- the mapping $a \mapsto \text{top}(x(a))$, where a is a node of T and $x(a)$ is the node of Y such that $a \in x(a)$; and
- the mapping $a \mapsto \rho(\text{path}_T(a, \text{top}(x(a))))$.

For $a \in V(T)$ we define $N^\uparrow(a)$ as before: $N^\uparrow(a)$ comprises all strict ancestors of a in T that are bound to a via functions present in $H_{\mathfrak{T}}$. We have the following analogue of Lemma 3.9.

Lemma 3.11. *For each $a \in V(T)$,*

$$\left| \{b \in V(T) : b \preceq_T a\} \cap \bigcup_{a' \succeq_T a} N^\uparrow(a') \right| \leq 2.$$

Proof. The only nodes that may be contained in the involved set are a^\uparrow and $\text{top}(x(a))$. \square

We may also prove the following analogue of Lemma 3.10.

Lemma 3.12. *Fix $c_0, c_1 \in [k]$. Then there formulas*

$$\{\zeta_{c_0, c_1, d_1, d_2}(p_0, p_1, q, q_0, q_1) : d_0, d_1 \in [k]\}$$

in the vocabulary of $H_{\mathfrak{T}}$ such that the following holds for all distinct $u_0, u_1 \in U$ satisfying $c_0 = \chi(u_0)$ and $c_1 = \chi(u_1)$, where $a_0 = \pi(u_0)$ and $a_1 = \pi(u_1)$. There is a unique 5-tuple $(d_0, d_1, t, t_0, t_1) \in [k]^2 \times V(T)^3$ such that $H_{\mathfrak{T}} \models \zeta_{c_0, c_1, d_0, d_1}(a_0, a_1, t, t_0, t_1)$:

- $t = \text{top}(\varpi(u_0) \wedge_Y \varpi(u_1))$;
- t_0 is the \preceq_T -maximum node of $\varpi(u_0) \wedge_Y \varpi(u_1)$ satisfying $t_0 \preceq_T \pi(u_0)$;
- t_1 is the \preceq_T -maximum node of $\varpi(u_0) \wedge_Y \varpi(u_1)$ satisfying $t_1 \preceq_T \pi(u_1)$;
- $d_0 = \kappa_{\mathfrak{T}}(u_0, t_0)$; and
- $d_1 = \kappa_{\mathfrak{T}}(u_1, t_1)$.

Proof. As in the proof of Lemma 3.10, we describe a mechanism of determining (d_0, d_1, t, t_0, t_1) from c_0, c_1, a_0, a_1 given access to $H_{\mathfrak{T}}$. It is straightforward to formulate this mechanism in first-order logic, which gives rise to the postulated formulas.

Let $x_0 = \varpi(u_0)$ and $x_1 = \varpi(u_1)$; note that x_0 and x_1 can be inferred from a_0 and a_1 . First, we check whether $x_0 = x_1$. If this is the case, then we have

- $t = \text{top}(x_0) = \text{top}(x_1)$;
- $t_0 = a_0$;
- $t_1 = a_1$;
- $d_0 = c_0$; and
- $d_1 = c_1$.

Otherwise, $x_0 \wedge_Y x_1$ is equal to the root r of Y . Then:

- $t = \text{top}(r)$ is the root of T ;
- $t_0 = a_0$ if $x_0 = r$, or t_0 is the parent of $\text{top}(x_0)$ in T otherwise;
- $t_1 = a_1$ if $x_1 = r$, or t_1 is the parent of $\text{top}(x_1)$ in T otherwise;
- $d_0 = c_0$ if $x_0 = r$, or $d_0 = \rho(e(\text{top}(x_0))) \circ \rho(\text{path}_T(a_0, \text{top}(x_0)))(c_0)$ otherwise; and
- $d_1 = c_1$ if $x_1 = r$, or $d_1 = \rho(e(\text{top}(x_1))) \circ \rho(\text{path}_T(a_1, \text{top}(x_1)))(c_1)$ otherwise.

This concludes the proof. \square

3.2.3 Completing the induction

We now utilize the understanding obtained in the previous sections to complete the proof of [Theorem 1.2](#) through an induction scheme. Let $\ell \leq 3k^k$ be the length of the sequence of classes provided by [Lemma 3.1](#).

Recall that we work with a k -NLC-tree $\mathfrak{T} = (T, U, \rho, \pi, \eta, \chi)$ generating G . We define a sequence of factorizations $\mathcal{Q}_1, \dots, \mathcal{Q}_\ell$ of \mathfrak{T} through backward induction as follows:

- \mathcal{Q}_ℓ consists of one factor, being the whole tree T itself; and
- for $i < \ell$, \mathcal{Q}_i is obtained from \mathcal{Q}_{i+1} by replacing each factor $F \in \mathcal{Q}_i$ with all the factors of $\mathcal{P}(\mathfrak{T}_F)$.

Thus, [Lemma 3.1](#) asserts that \mathcal{Q}_1 is a factorization of \mathfrak{T} into single-node factors.

Next, for each $i \in [\ell]$ and factor $F \in \mathcal{Q}_i$ we define a structure J_F . Intuitively, J_F encodes the structure $H_{\mathfrak{T}_F}$ that we defined in the previous section, as well as all the structures $J_{F'}$ for $F' \in \mathcal{P}(\mathfrak{T}_F)$, constructed in the previous step of the induction. Thus, the universe of \mathfrak{T}_F is $V(F)$, while the relations in \mathfrak{T}_F are defined by induction on i as follows.

For $i = 1$, the tree F has exactly one node, say a . Structure J_F stores only the value $\eta(a)$, encoded using unary relations on a .

For $i > 1$, the structure J_F is constructed as a superposition of the structure $H_{\mathfrak{T}_F}$ and structures $J_{F'}$ for $F' \in \mathcal{P}(\mathfrak{T}_F)$ as follows. First, consider the induced k -NLC-tree \mathfrak{T}_F and construct the structure $H_{\mathfrak{T}_F}$ for it as in the previous section. This structure has $V(F)$ as its universe. Next, for each factor $F' \in \mathcal{P}(\mathfrak{T}_F)$, consider the structure $J_{F'}$ constructed in the previous step of induction and add all the tuples from all the relations of $J_{F'}$ to J_F . While doing this, we reuse relation names: we assume that all the structures $J_{F'}$ are over the same vocabulary, so to obtain a relation R from this vocabulary in J_F we take the union of relations R taken from structures $J_{F'}$ for $F' \in \mathcal{P}(\mathfrak{T}_F)$. Note here that the universes of structures $J_{F'}$ are pairwise disjoint, and the vocabulary used for encoding $H_{\mathfrak{T}_F}$ is assumed to be disjoint from the vocabulary used for encoding structures $J_{F'}$. Finally, for technical reasons we add to J_F a function $\text{root}_i(\cdot)$ that maps each node $a \in V(F)$ to the root of F .

Let now $J_{\mathfrak{T}} := J_T$, where T is the unique factor of \mathcal{Q}_ℓ . Further, let $J_{\mathfrak{T}}^*$ be the structure obtained from $J_{\mathfrak{T}}$ by adding U to the universe, together with unary and binary relations encoding mappings $u \mapsto \pi(u)$ and $u \mapsto \chi(u)$, for $u \in U$.

First, we verify that $J_{\mathfrak{T}}^*$ contains all the information needed to reconstruct G .

Lemma 3.13. *There is a first-order formula $\alpha(p_0, p_1)$ over the vocabulary of $J_{\mathfrak{T}}^*$ such that for all $u_0, u_1 \in U$, we have $J_{\mathfrak{T}}^* \models \alpha(u_0, u_1)$ if and only if $u_0 u_1 \in E(G)$.*

Proof. For a pair of vertices $a_0, a_1 \in V(T)$, let the *level* of (a_0, a_1) be the smallest integer i such that a_0 and a_1 belong to the same factor of \mathcal{Q}_i . As \mathcal{Q}_ℓ consists of one factor – the whole tree T – the level of every pair is upper bounded by ℓ . We shall inductively define formulas $\beta_{c_0, c_1}^i(p_0, p_1)$ for $c_0, c_1 \in [k]$ and $i \in [\ell]$ satisfying the following property: for every pair $(a_0, a_1) \in V(T)^2$ of level at most i , if there are vertices $u_0, u_1 \in U$ satisfying $\pi(u_0) = a_0$, $\pi(u_1) = a_1$, $\chi(u_0) = c_0$, and $\chi(u_1) = c_1$, then $J_{\mathfrak{T}}^* \models \beta_{c_0, c_1}^i(a_0, a_1)$ iff $u_0 u_1 \in E(G)$. If we succeed in this, then formula $\alpha(u_0, u_1)$ can be written by first defining $a_0 = \pi(u_0)$, $a_1 = \pi(u_1)$, $c_0 = \chi(u_0)$, and $c_1 = \chi(u_1)$, and then applying $\beta_{c_0, c_1}^\ell(a_0, a_1)$.

Consider first the base case $i = 1$. As factorization \mathcal{Q}_1 places every node of T in a different factor, then condition that (a_0, a_1) has level at most 1 boils down to $a_0 = a_1$. Hence $\beta_{c_0, c_1}^1(a_0, a_1)$ only needs to check that $a_0 = a_1$ and that $(c_0, c_1) \in \eta(a_0)$.

We proceed to the induction step. Let F be the factor of \mathcal{Q}_i that contains both a_0 and a_1 . We shall assume that the quotient tree $\mathfrak{T}_F/\mathcal{P}(\mathfrak{T}_F)$ is splendid, hence we will use formulas provided by [Lemma 3.12](#) for the k -NLC-tree \mathfrak{T}_F . Note here that the structure $H_{\mathfrak{T}_F}$ encoding \mathfrak{T}_F is contained in $J_{\mathfrak{T}}^*$. Hence, these formulas may be applied in $J_{\mathfrak{T}}^*$ in the same manner as in $H_{\mathfrak{T}_F}$, provided that we appropriately relativize them to the elements of $V(F)$; these can be distinguished as elements mapped to the root of F by $\text{root}_i(\cdot)$. The reasoning in the other case, when $\mathfrak{T}_F/\mathcal{P}(\mathfrak{T}_F)$ is shallow, proceeds in the same way and is even simpler, as we may use [Lemma 3.12](#) instead of [Lemma 3.10](#).

We first check whether $\varphi_{c_0, c_1}(a_0, a_1)$ holds in $H_{\mathfrak{T}_F}$. If this is the case, then we may immediately determine whether u_0 and u_1 are adjacent in G by checking whether $\psi_{c_0, c_1}(a_0, a_1)$ holds in $H_{\mathfrak{T}_F}$. Otherwise, using formulas $\zeta_{c_0, c_1, d_0, d_1}(p_0, p_1, q, q_0, q_1)$ we can find suitable colors $d_0, d_1 \in [k]$ and nodes $t, t_0, t_1 \in V(F)$, as described in [Lemma 3.10](#). Note here that if F' is the factor of $\mathcal{P}(\mathfrak{T}_F)$ that contains the least common ancestor of a_0 and a_1 , then

- $t = \text{top}(F')$;
- $t_0 = \pi_{F'}(u_0)$;
- $t_1 = \pi_{F'}(u_1)$;
- $d_0 = \chi_{F'}(u_0)$; and
- $d_1 = \chi_{F'}(u_1)$.

Hence, to decide whether $u_0 u_1 \in E(G)$, it suffices to check whether $J_{\mathfrak{T}}^* \models \alpha_{d_0, d_1}^{i-1}(t_0, t_1)$, which is a formula that we constructed in the previous step of induction. \square

Recall that the *Gaifman graph* of a structure A is the undirected graph $\text{Gaif}(A)$ whose vertex set is the universe of A , and where two elements are considered adjacent if and only if they appear simultaneously in a tuple in a relation in A . Define

$$\mathcal{D} := \{ \text{Gaif}(J_{\mathfrak{T}}^*) : \mathfrak{T} \text{ is a } k\text{-NLC-tree generating a graph from } \mathcal{C} \}.$$

That the class \mathcal{D} has bounded treewidth is then proved using the characterization of treewidth through the *strong reachability* relation, with the help of [Lemma 3.9](#) and [Lemma 3.11](#).

For the proof of [Lemma 3.14](#), we need several definitions.

Let G be a graph and let \leq be a *vertex ordering* of G , that is, a linear order on the vertex set of G . For a vertex u and v of G , we say that v is *strongly reachable* from u in \leq if $v \leq u$ and in G there exists a path P from u to v such that $u < w$ for every internal vertex w of P . Then, we define the *strong reachability set* of u , denoted $\text{SReach}_{\infty}[G, \leq, u]$ as the set of all vertices of G that are strongly reachable from u in \leq . The *strong ∞ -coloring number* of G is defined as

$$\text{scol}_{\infty}(G) = \min_{\leq} \max_{u \in V(G)} |\text{SReach}_{\infty}[G, \leq, u]|,$$

where the minimum ranges over all vertex orderings of G . It is folklore that the strong ∞ -coloring number essentially coincides with treewidth.

Theorem 3.1 (see e.g. Chapter 1, Theorem 1.19 of [\[37\]](#)). *For every graph G , the treewidth of G is equal to $\text{scol}_{\infty}(G) - 1$.*

We now use [Theorem 3.1](#) together with [Lemma 3.9](#) and [Lemma 3.11](#) to prove the following.

Lemma 3.14. *For every graph $G \in \mathcal{D}$, the treewidth of G is at most $3k^k \cdot (836h + 2k + 4)$.*

Proof. By [Theorem 3.1](#), it suffices to give a vertex ordering of G where each strong reachability set has size at most $3k^k \cdot (836h + 2k + 4) + 1$. Let $G = \text{Gaif}(J_{\mathfrak{T}}^*)$, where $\mathfrak{T} = (T, U, \rho, \pi, \eta, \chi)$ is a k -NLC-tree that generates a graph from \mathcal{C} . Then $V(G) = U \cup V(T)$. Let \leq be a vertex ordering of G constructed as follows: first put all the nodes of T in any order that extends \preceq_T (that is, $u \preceq_T v$ entails $u \leq v$), and then put all the vertices of U in any order. Our goal is to establish an upper bound on the sizes of strong reachability sets with respect to the ordering \leq .

Observe that for $u \in U$, we have $\text{SReach}_\infty[G, \leq, u] = \{u, \pi(u)\}$, so this is a set of size 2. Consider then any $a \in V(T)$. From the construction of $J_{\mathfrak{T}}$ it follows that all the edges of G which connect two nodes $V(T)$ in fact connect a node of T with its ancestor. Hence, we have

$$\text{SReach}_\infty[G, \leq, a] \subseteq \{a\} \cup \bigcup_{i=1}^{\ell} N_i^\uparrow(a),$$

where $N_i^\uparrow(a)$ is the set $N^\uparrow(a)$ evaluated in the structure $H_{\mathfrak{T}_{F_i}}$, where F_i is the factor from \mathcal{Q}_i that contains a . By [Lemma 3.9](#) and [Lemma 3.11](#), each of the sets $N_i^\uparrow(a)$ has size at most $836h + 2k + 4$, so

$$|\text{SReach}_\infty[G, \leq, a]| \leq \ell \cdot (836h + 2k + 4) + 1 = 3k^k \cdot (836h + 2k + 4) + 1,$$

as required. \square

The bound obtained in [Lemma 3.14](#) is not optimal, and could be easily reduced. Note that it is not known whether there is a collapse in the hierarchy of classes with bounded treewidth with respect to first-order transductions, that is, whether there exist integers $k < k'$ with the property that the class of graphs with treewidth at most k' can be transduced from the class of graphs with treewidth at most k . We conjecture that this is not the case.

We are now able to prove [Theorem 1.2](#), which we restate below.

Theorem 1.2. *If a class of graphs \mathcal{C} has bounded rankwidth, then the following conditions are equivalent:*

1. \mathcal{C} has a stable edge relation;
2. \mathcal{C} is stable;
3. \mathcal{C} is monadically stable;
4. \mathcal{C} is a transduction of a class with bounded treewidth.

Proof. For a graph G , let \widehat{G} be the graph obtained from G by subdividing every edge uv twice, that is, replacing it with a path $u - s_{uv}^u - s_{uv}^v - v$. Let $\widehat{\mathcal{D}} = \{\widehat{G} : G \in \mathcal{D}\}$. As subdividing edges does not increase the treewidth and \mathcal{D} has bounded treewidth by [Lemma 3.14](#), the same bound also applies to $\widehat{\mathcal{D}}$.

We now prove that there is a transduction from $\widehat{\mathcal{D}}$ onto \mathcal{C} , hence establishing the only non-trivial implication of the theorem.

Consider any graph $G \in \mathcal{C}$. Let \mathfrak{T} be any k -NLC-tree that generates G . Let $M = \text{Gaif}(J_{\mathfrak{T}}^*)$. We argue that G can be transduced from $\widehat{M} \in \widehat{\mathcal{D}}$ using a fixed transduction that depends only on k .

We first argue that the structure $J_{\mathfrak{T}}^*$ can be transduced from \widehat{M} . First, we add colors to distinguish the original vertices of M from the subdividing vertices (i.e. vertices s_{uv}^u and s_{uv}^v introduced when constructing \widehat{M} from M). Now, recall that the vocabulary of $J_{\mathfrak{T}}^*$ consists only of unary relations and partial functions. Unary relations present in $J_{\mathfrak{T}}^*$ can be introduced directly. For every partial function f present in $J_{\mathfrak{T}}^*$, we transduce it as follows. First, we introduce a unary predicate Z_f which selects vertices $s_{u f(u)}^u$ for u ranging over the domain of f . Then it is straightforward to interpret f using a first-order formula involving Z_f . Thus, we have introduced all the relations present in $J_{\mathfrak{T}}^*$, and it remains to use a universe restriction formula to dispose of all the subdividing vertices, which should not be included in the universe of $J_{\mathfrak{T}}^*$.

Now that $J_{\mathfrak{T}}^*$ has been transduced from \widehat{M} , we can use formula $\alpha(p_0, p_1)$ provided by [Lemma 3.13](#) to interpret the edge relation of G in $J_{\mathfrak{T}}^*$. Restricting the universe to U finishes the construction of G from \widehat{M} by means of a transduction. \square

Finally, let us discuss the algorithmic aspects of the proof. Given a graph $G \in \mathcal{C}$, we can compute a k -NLC-tree generating G in cubic time [[36](#)], for some constant k . The hierarchical factorization provided by [Lemma 3.1](#) can be computed in polynomial time, because the result of Colcombet [[7](#)] is effective. It is straightforward to see that all the further elements of the construction, like determining

the types, partitioning into blocks, etc., which amount to the construction of the structure $J_{\mathcal{C}}^*$, can be carried out in polynomial time. Thus, given $G \in \mathcal{C}$, we can in polynomial time compute a graph of bounded treewidth H from which G can be transduced, together with a suitable monadic extension of H . The interpretation yielding G from this monadic extension of H can be computed as well.

4 Some combinatorial consequences of Theorem 1.2

Theorem 1.2 asserts that each class with bounded rankwidth and stable edge relation is a transduction of a class with bounded treewidth. We now derive some consequences of this result.

Classes with bounded treewidth are examples of classes with bounded expansion [32]. Recall that a class \mathcal{C} has *bounded expansion* if there exists a function $f: \mathbb{N} \rightarrow \mathbb{N}$ with the property that every graph H such that a subdivision of H with edges subdivided at most r times is a subgraph of a graph in \mathcal{C} has average degree at most $f(r)$. (The reader is referred to [33] for an in-depth study of these classes.)

These classes are characterized by the existence of special covers. Let complexity be a graph parameter, such as treewidth or rankwidth. A class \mathcal{C} has *low complexity covers* if for each positive integer p there exists a constant C_p and a class \mathcal{X}_p with bounded complexity, such that each graph $G \in \mathcal{C}$ can be covered by C_p induced subgraphs $H_1, \dots, H_{C_p} \in \mathcal{X}_p$ in such a way that every subset of p vertices of G are jointly covered by some H_i ($1 \leq i \leq C_p$).

Recall that the *treedepth* of a graph G [33] is the minimum number of levels of a rooted forest Y such that G is a subgraph of the ancestor-descendant closure of Y . Equivalently, the treedepth of a graph G is the minimum clique number of a supergraph of G that is a trivially perfect graph. The following result follows from the characterization of bounded expansion in terms of low treedepth colorings.

Theorem 4.1 ([32]). *A class has bounded expansion if and only if it has low treedepth covers.*

An extension of this result gives a characterization of the graph classes that are transductions of classes with bounded expansion. Following [18], we say that such classes have *structurally bounded expansion*.

Theorem 4.2 ([18]). *A class has structurally bounded expansion if and only if it has low shrubdepth covers.*

Recall that a class \mathcal{S} has *bounded shrubdepth* if there exist constants m and h such that for every graph $G \in \mathcal{S}$ there is a rooted tree Y with set of leaves $L(Y) = V(G)$, a coloring $c: L(Y) \rightarrow [m]$ and an assignment $v \mapsto f_v$ of a symmetric function $f_v: [m] \times [m] \rightarrow \{0, 1\}$ to each internal node v of Y , in such a way that two vertices $u, v \in V(G)$ are adjacent in G if and only if $f_{u \wedge_Y v}(c(u), c(v)) = 1$ [20, 19]. In particular, the subgraph of G induced by each single color class is a cograph. Since cographs are perfect, in particular we have $\chi(G) \leq m \omega(G)$. We deduce the following corollary of Theorem 4.2.

Corollary 4.1. *For every structurally bounded expansion class \mathcal{C} there exists a constant C such that the vertex set of every $G \in \mathcal{C}$ can be partitioned into at most C classes, each inducing a cograph.*

In particular, every structurally bounded expansion class is linearly χ -bounded.

Note that a class has bounded shrubdepth if and only if it can be transduced from a class with bounded treedepth [20].

In an effort to generalize low treedepth coverings further, classes with low rankwidth covers have been studied in [29]. As a direct consequence of Theorem 1.2 and Corollary 4.1, we have:

Theorem 1.4. *Every class with low rankwidth covers and stable edge relation is linearly χ -bounded.*

Proof. Let \mathcal{C} be the class in question. Taking $p = 1$ in the definition, for every graph $G \in \mathcal{C}$, we can partition the vertex set of G into a bounded number of parts, each of which induces a subgraph that belongs to a class \mathcal{D} that has bounded rankwidth and a stable edge relation. By [Theorem 1.2](#), \mathcal{D} can be transduced from a class of bounded treewidth, hence it has structurally bounded expansion. By [Corollary 4.1](#) we conclude that \mathcal{D} is linearly χ -bounded, so it follows that \mathcal{C} is linearly χ -bounded as well. \square

It is known that the chromatic number of graphs with (linear) cliquewidth at most k cannot be computed in $f(k)n^{2^{o(k)}}$ time for any computable function f , unless ETH fails [\[21\]](#). However, it follows from what precedes that for each class \mathcal{C} with bounded rankwidth and stable edge relation there is an $\mathcal{O}(n^3)$ -time algorithm, which gives a constant factor approximation for the chromatic number. Indeed, given a graph G from the considered class, we can first use the result of Oum and Seymour [\[36\]](#) to compute in cubic time a k -NLC-tree of G for some constant k (or any equivalent decomposition, such as a clique expression). Then, using standard dynamic programming we can compute the clique number of the graph in linear time. By [Theorem 1.4](#), this clique number is a constant-factor approximation of the chromatic number.

We also deduce the following result.

Theorem 1.5. *A class has low rankwidth covers and a stable edge relation if and only if it is a transduction of a class with bounded expansion.*

Proof. If a class has structurally bounded expansion, then it has low shrubdepth covers [\[16\]](#), which are special instances of low rankwidth covers. Moreover, as bounded expansion classes are nowhere dense, they are monadically stable [\[1\]](#), hence structurally bounded expansion classes have a stable edge relation.

Conversely, assume a class \mathcal{C} has low rankwidth covers and stable edge relation. Then for each integer p there exists a constant C_p and a class \mathcal{R}_p with bounded rankwidth such that each graph $G \in \mathcal{C}$ can be covered by C_p induced subgraphs $H_1, \dots, H_{C_p} \in \mathcal{R}_p$ in such a way that every subset of p vertices of G are jointly covered by some H_i ($1 \leq i \leq C_p$). As \mathcal{C} has a stable edge relation, it excludes some half-graph F . Obviously, we can require that \mathcal{R}_p contains only induced subgraphs of graphs in \mathcal{C} . Thus graphs in \mathcal{R}_p exclude F as well, so \mathcal{R}_p has a stable edge relation. By [Theorem 1.2](#), \mathcal{R}_p can be transduced from a class with bounded treewidth, hence \mathcal{R}_p has structurally bounded expansion. It follows from [Theorem 4.2](#) that there exists C'_p and a class \mathcal{T}_p with bounded shrubdepth such that each graph H_i can be covered by C'_p induced subgraphs $T_{i,1}, \dots, T_{i,C'_p} \in \mathcal{T}_p$ in such a way that every subset of p vertices of H_i are jointly covered by some $T_{i,j}$. We deduce that \mathcal{C} has low shrubdepth covers, so it has structurally bounded expansion. \square

In [\[18\]](#), it was stressed that one of the main difficulties arising when considering low shrubdepth covers of structurally bounded expansion classes (whose existence is asserted in [Theorem 4.2](#)) is that we do not know if they may be computed in polynomial time (and that polynomial-time computation of these covers for $p = 2$ ensures that FO-model checking is FPT on the class). A consequence of this paper is that for a class with structurally bounded treewidth (that is, a class with bounded rankwidth and stable edge relation), and for each integer p , low shrubdepth covers with parameter p can be computed in polynomial time. Such a property also holds for structurally bounded degree classes (that is, transductions of classes with bounded degree) [\[16\]](#), as well as classes obtained from bounded expansion classes by a transduction consisting a bounded number of subgraph complementations [\[17\]](#). We conjecture that this holds in general.

Conjecture 4.1. *For every structurally bounded expansion class \mathcal{C} , computing a low shrubdepth cover of a graph $G \in \mathcal{C}$ at depth p is fixed parameter tractable when parameterized by p .*

5 Monadic dependence meets stability

In this section we prove [Theorem 1.3](#), which shows that the equivalence of the first three conditions of [Theorem 1.2](#) (and [Theorem 1.1](#)) is in fact a more general phenomenon that occurs in every monadically dependent graph class. In our proof, we shall need the following classical theorem.

Theorem 5.1 (Canonical Ramsey Theorem [14]). *For every integer n there exists an integer N with the following property: Suppose that all pairs (a, b) of integers with $1 \leq a < b \leq N$ are arbitrarily distributed into classes. Then there is an increasing sequence of integers $1 \leq x_1 < x_2 < \dots < x_n \leq N$ such that one of the following four sets of conditions holds, where it is assumed that $1 \leq \alpha < \beta \leq n$; $1 \leq \gamma < \delta \leq n$:*

1. All (x_α, x_β) belong to the same class.
2. (x_α, x_β) and (x_γ, x_δ) belong to the same class if, and only if, $\alpha = \gamma$.
3. (x_α, x_β) and (x_γ, x_δ) belong to the same class if, and only if, $\beta = \delta$.
4. (x_α, x_β) and (x_γ, x_δ) belong to the same class if, and only if, $\alpha = \gamma$; $\beta = \delta$.

Let us now proceed to the proof of [Theorem 1.3](#), restated below.

Theorem 1.3. *For a monadically dependent graph class \mathcal{C} , the following conditions are equivalent:*

1. \mathcal{C} has a stable edge relation;
2. \mathcal{C} is stable;
3. \mathcal{C} is monadically stable.

Proof. Implications $3 \Rightarrow 2 \Rightarrow 1$ are obvious, so it remains to prove the following: if a class \mathcal{C} is monadically dependent but also monadically unstable, then in fact \mathcal{C} has an unstable edge relation. Hence, assume that \mathcal{C} is monadically unstable. In the following, we write $\binom{[n]}{2}$ for the set of all pairs of integers (i, j) such that $1 \leq i < j \leq n$.

A formula $\alpha(\bar{x})$ is *functional* on a class if there is a variable $x \in \bar{x}$ such that for every G in the class and $u \in V(G)$, there exists at most one tuple $\bar{u} \in V(G)^{\bar{x}}$ such that $G \models \alpha(\bar{u})$ and $\bar{u}(x) = u$. We shall say that a triple of formulas $\tau = (\alpha(\bar{x}), \beta(\bar{y}), \eta(\bar{x}, \bar{y}))$ in a monadic vocabulary of graphs is *problematic* if there exists a monadic expansion \mathcal{C}^+ of \mathcal{C} , whose vocabulary contains the vocabularies of α , β , and η , such that α and β are functional on \mathcal{C}^+ , and for every $n \in \mathbb{N}$ there exists $G \in \mathcal{C}^+$ and tuples $\bar{a}_1, \dots, \bar{a}_n \in V(G)^{\bar{x}}$ and $\bar{b}_1, \dots, \bar{b}_n \in V(G)^{\bar{y}}$ satisfying the following:

- for all $i \in [n]$ we have $G \models \alpha(\bar{a}_i)$ and $G \models \beta(\bar{b}_i)$; and
- for all $(i, j) \in \binom{[n]}{2}$ we have $G \models \eta(\bar{a}_i, \bar{b}_j)$ and $G \models \neg\eta(\bar{a}_j, \bar{b}_i)$.

Note that we do not specify whether $\eta(\bar{a}_i, \bar{b}_i)$ should hold or not in G . The pair of sequences $\bar{a}_1, \dots, \bar{a}_n$ and $\bar{b}_1, \dots, \bar{b}_n$ as above shall be called a τ -ladder of length n in G . Observe that if in graphs from \mathcal{C}^+ one can find arbitrarily long τ -ladders, then η is unstable on \mathcal{C}^+ .

As \mathcal{C} is monadically unstable, by [Theorem 2.2](#) we know that there is a transduction from \mathcal{C} onto the class of all finite half-graphs. By the definition of a transduction, this implies that there exists a monadic expansion \mathcal{C}^+ of \mathcal{C} and a formula $\varphi(x, y)$ with two free variables x and y such that φ is unstable on \mathcal{C}^+ . By taking $\alpha(x)$ and $\beta(y)$ to be true formulas, we conclude the following.

Claim. *There exists a problematic triple of formulas.*

We now investigate the properties of problematic formulas.

Claim. *If $\tau = (\alpha(\bar{x}), \beta(\bar{y}), \eta(\bar{x}, \bar{y}))$ is problematic, then so is $\tau' = (\alpha(\bar{x}), \beta(\bar{y}), \neg\eta(\bar{x}, \bar{y}))$.*

Proof of the claim. It suffices to observe that reversing both sequences in a τ -ladder yields a τ' -ladder. \triangleleft

Claim. *If the triple $\tau = (\alpha(\bar{x}), \beta(\bar{y}), \eta_1(\bar{x}, \bar{y}) \vee \eta_2(\bar{x}, \bar{y}))$ is problematic, then at least one of the triples $\tau_1 = (\alpha(\bar{x}), \beta(\bar{y}), \eta_1(\bar{x}, \bar{y}))$ and $\tau_2 = (\alpha(\bar{x}), \beta(\bar{y}), \eta_2(\bar{x}, \bar{y}))$ is problematic.*

Proof of the claim. By assumption, there is a monadic expansion \mathcal{C}^+ of \mathcal{C} such that there are arbitrarily long τ -ladders in graphs from \mathcal{C} . Suppose $\bar{a}_1, \dots, \bar{a}_n$ and $\bar{b}_1, \dots, \bar{b}_n$ is such a τ -ladder in some $G \in \mathcal{C}$. Observe that for all $(i, j) \in \binom{[n]}{2}$, we have $G \models \eta_1(\bar{a}_i, \bar{b}_j)$ or $G \models \eta_2(\bar{a}_i, \bar{b}_j)$. By Ramsey's theorem and since n can be chosen arbitrarily large, by restricting attention to a sub-ladder we may assume that one of these cases holds for every pair $(i, j) \in \binom{[n]}{2}$, say the first one by symmetry. However, for all $(i, j) \in \binom{[n]}{2}$ we also have $G \models \neg(\eta_1(\bar{a}_j, \bar{b}_i) \vee \eta_2(\bar{a}_j, \bar{b}_i))$, which implies $G \models \neg\eta_1(\bar{a}_j, \bar{b}_i)$. We conclude that $\bar{a}_1, \dots, \bar{a}_n$ and $\bar{b}_1, \dots, \bar{b}_n$ form a τ_1 -ladder of length n . As n can be chosen arbitrarily large, τ_1 is problematic. \triangleleft

Claim. *If a triple $\tau = (\alpha(\bar{x}), \beta(\bar{y}), \eta(\bar{x}, \bar{y}))$ is problematic and $\eta(\bar{x}, \bar{y}) = \exists z \zeta(\bar{x}, \bar{y}, z)$, then there is a problematic triple of the form $\tau' = (\alpha'(\bar{x}'), \beta'(\bar{y}'), \zeta(\bar{x}', \bar{y}'))$ where either $(\bar{x}', \bar{y}') = (\bar{x} \cup \{z\}, \bar{y})$ or $(\bar{x}', \bar{y}') = (\bar{x}, \bar{y} \cup \{z\})$.*

Proof of the claim. Consider any $n \in \mathbb{N}$ and let N be the integer given by the Canonical Ramsey Theorem (Theorem 5.1) for n . By assumption, there is a monadic expansion \mathcal{C}^+ of \mathcal{C} such that there exist arbitrarily long τ -ladders in graphs from \mathcal{C}^+ . Hence, we can find a τ -ladder $\bar{a}_1, \dots, \bar{a}_{2N}, \bar{b}_1, \dots, \bar{b}_{2N}$ of length $2N$ in some $G \in \mathcal{C}^+$. By restricting attention to a sub-ladder consisting of every odd element of the sequence $\bar{a}_1, \dots, \bar{a}_{2N}$ and every even element of the sequence $\bar{b}_1, \dots, \bar{b}_{2N}$, and appropriately reindexing, we find a τ -ladder $\bar{a}_1, \dots, \bar{a}_N, \bar{b}_1, \dots, \bar{b}_N$ of length- N in G such that $G \models \eta(\bar{a}_i, \bar{b}_i)$ for all $i \in [n]$. Note that tuples $\bar{a}_1, \dots, \bar{a}_N$ have to be pairwise different, because for each $i \in [N]$, the smallest $j \in [N]$ satisfying $G \models \eta(\bar{a}_i, \bar{b}_j)$ is equal to i . Similarly, tuples $\bar{b}_1, \dots, \bar{b}_N$ have to be pairwise different as well.

Let $x \in \bar{x}$ and $y \in \bar{y}$ be the variables witnessing that α and β are functional, respectively. For $i \in [n]$, let $a_i = \bar{a}_i(x)$ and $b_i = \bar{b}_i(y)$. As α is functional, we conclude that vertices a_1, \dots, a_N are pairwise different, and similarly vertices b_1, \dots, b_N are pairwise different as well. Let $A = \{a_1, \dots, a_N\}$ and $B = \{b_1, \dots, b_N\}$, and let G^{AB} be a monadic expansion of G where A and B are additionally distinguished using unary predicates, which we shall respectively call A and B by a slight abuse of notation.

Let \prec be the (strict) lexicographic order on $\binom{[N]}{2}$. Observe that there exists a formula $\lambda(\bar{x}, \bar{y}, \bar{x}^\circ, \bar{y}^\circ)$, where \bar{x}° and \bar{y}° are copies of \bar{x} and \bar{y} , respectively, such that the following holds: if $(\bar{a}, \bar{b}) = (\bar{a}_i, \bar{b}_j)$ and $(\bar{a}^\circ, \bar{b}^\circ) = (\bar{a}_{i^\circ}, \bar{b}_{j^\circ})$ for some $(i, j), (i^\circ, j^\circ) \in \binom{[n]}{2}$, then $G^+ \models \lambda(\bar{a}, \bar{b}, \bar{a}^\circ, \bar{b}^\circ)$ if and only if $(i, j) \prec (i^\circ, j^\circ)$. Indeed, the formula $\forall \bar{w} [(B(\bar{w}) \wedge \beta(\bar{w}) \wedge \eta(\bar{a}^\circ, \bar{w})) \rightarrow \eta(\bar{a}, \bar{w})]$ (where the variable $w \in \bar{w}$ corresponds to the variable $y \in \bar{y}$) allows us to check the assertion $i \leq i^\circ$. A formula expressing $j \leq j^\circ$ can be written in a symmetric way. Then the condition $(i, j) \prec (i^\circ, j^\circ)$ can be expressed using a boolean combination of assertions $i \leq i^\circ, i \geq i^\circ, j \leq j^\circ$, and $j \geq j^\circ$.

As for every pair $(i, j) \in \binom{[N]}{2}$ we have $G \models \exists z \zeta(\bar{a}_i, \bar{b}_j, z)$, there is a vertex $c \in V(G)$ such that $G \models \zeta(\bar{a}_i, \bar{b}_j, c)$. Let C be an inclusion-wise minimal subset of $V(G)$ such that for each $(i, j) \in \binom{[N]}{2}$ there exists $c \in C$ satisfying $G \models \zeta(\bar{a}_i, \bar{b}_j, c)$. For every $c \in C$, define

$$J(c) = \left\{ (i, j) \in \binom{[N]}{2} : \zeta(\bar{a}_i, \bar{b}_j, c) \right\}.$$

Note that by the minimality of C , the sets $J(c)$ are pairwise not contained in one another. Let G^{ABC} be the monadic expansion of G^{AB} where C is additionally distinguished using a unary predicate C .

Now, for $c, c' \in C$, we set

$$c \sqsubset c' \quad \text{if and only if} \quad \min_{\prec} (J(c) \setminus J(c')) \prec \min_{\prec} (J(c') \setminus J(c)).$$

It is straightforward to see that \sqsubset is a (strict) linear order on C . Let us partition pairs $(i, j) \in \binom{[N]}{2}$ into classes $\{I(c) : c \in C\}$ as follows:

$$(i, j) \in I(c) \quad \text{if and only if} \quad c = \min_{\sqsubset} \{d \in C : (i, j) \in J(d)\}.$$

Using the formula λ we can easily write a formula $\kappa(\bar{x}, \bar{y}, z)$ with the following property: for all $(i, j) \in \binom{[N]}{2}$ and $c \in C$, we have $G^{ABC} \models \kappa(\bar{a}_i, \bar{b}_j, c)$ if and only if $(i, j) \in I(c)$.

By the Canonical Ramsey Theorem (Theorem 5.1) there exists $F \subseteq [N]$ such that $|F| = n$ and one of the following conditions is satisfied:

- (1) all pairs $(i, j) \in \binom{F}{2}$ belong to the same class $I(c)$, for some $c \in C$;
- (2) there exist pairwise different c_i such that $(i, j) \in I(c_i)$ for all $(i, j) \in \binom{F}{2}$;
- (3) there exist pairwise different c_j such that $(i, j) \in I(c_j)$ for all $(i, j) \in \binom{F}{2}$;
- (4) there exist pairwise different $c_{i,j}$ such that $(i, j) \in I(c_{i,j})$ for all $(i, j) \in \binom{F}{2}$.

Let $G^{A'B'C}$ be the monadic expansion of G^{ABC} where sets $A' = \{a_i : i \in F\}$ and $B' = \{b_i : i \in F\}$ are additionally distinguished using unary predicates A' and B' .

We first consider the second case above. Let $G^{A'B'CD}$ be a monadic expansion of $G^{A'B'C}$ that distinguishes the single vertex $b_{\max F}$ using a unary predicate D . Consider the formula

$$\alpha'(\bar{x}, z) = A'(x) \wedge C(z) \wedge \alpha(\bar{x}) \wedge \exists \bar{y} [D(y) \wedge \beta(\bar{y}) \wedge \kappa(\bar{x}, \bar{y}, z)].$$

Observe that for any $\bar{u} \in V(G)^{\bar{x}}$ and $w \in V(G)$, we have $G^{A'B'CD} \models \alpha'(\bar{u}, w)$ if and only if $\bar{u} = \bar{a}_i$ for some $i \in F$ and $w = c_i$. As $\alpha(\bar{x})$ is functional, it follows that so is $\alpha'(\bar{x}, z)$. It is now straightforward to see that $\{\bar{a}_i, c_i : i \in F\}$ and $\{\bar{b}_i : i \in F\}$ form a τ' -ladder in $G^{A'B'CD}$ of length n , where $\tau' = (\alpha'(\bar{x}, z), \beta(\bar{y}), \zeta(\bar{x}, \bar{y}, z))$. Hence, if the second case occurs for infinitely many n , then τ' is problematic.

The same argument applies if the first case occurs for infinitely many n , and a symmetric argument applies when the third case occurs for infinitely many n . We are left with considering the situation where the fourth case occurs for infinitely many n . Let $S = \{c_{i,j} : (i, j) \in \binom{F}{2}\}$. Observe that if we choose any subset $P \subseteq S$ and distinguish it using a unary predicate P in a monadic expansion $G^{A'B'CP}$ of $G^{A'B'C}$, then the formula

$$\xi(\bar{x}, \bar{y}) = A'(x) \wedge \alpha(\bar{x}) \wedge B'(y) \wedge \beta(\bar{y}) \wedge \exists z [P(z) \wedge \kappa(\bar{x}, \bar{y}, z)],$$

is true exactly for those tuples \bar{a}_i and \bar{b}_j for which $(i, j) \in \binom{F}{2}$ and $c_{i,j} \in P$. Hence, using ξ and different choices of P we may interpret in graphs $G^{A'B'CP}$ all subgraphs of a half-graph of order n . It follows that there is a transduction from \mathcal{C} onto the class of all bipartite graphs; this contradicts the assumption that \mathcal{C} is monadically dependent. \triangleleft

By the above claims we infer that there is a problematic triple $(\alpha(\bar{x}), \beta(\bar{y}), \eta(\bar{x}, \bar{y}))$ such that $\eta(\bar{x}, \bar{y})$ is an atomic formula. In particular, this means that there is a monadic expansion \mathcal{C}^+ of \mathcal{C} such that η is unstable on \mathcal{C}^+ . Since η is atomic, it is of one of the following forms: a unary predicate applied to any variable; the equality relation applied to any pair of variables; or the edge relation $E(\cdot, \cdot)$ applied to any pair of variables. The first two cases cannot happen, as such formulas are stable on every class of graphs. We conclude that the last case occurs, hence \mathcal{C} has an unstable edge relation. \square

6 Conclusion and Perspectives

We have started to explore the theory of monadic dependence and monadic stability from a graph theoretical point of view. Several interesting questions and conjectures arise from our studies. To put our research in perspective, we show in Figure 5 the following extended semi-lattice of property inclusions.

A quick examination of the figure reveals an unresolved question of prime importance. While Theorem 1.1 and Theorem 1.2 exactly identify classes of structurally bounded pathwidth/treewidth as monadically stable classes that have bounded (linear) rankwidth, the chart does not specify the alignment of *structurally nowhere dense* classes (i.e. transductions of nowhere dense classes). Clearly,

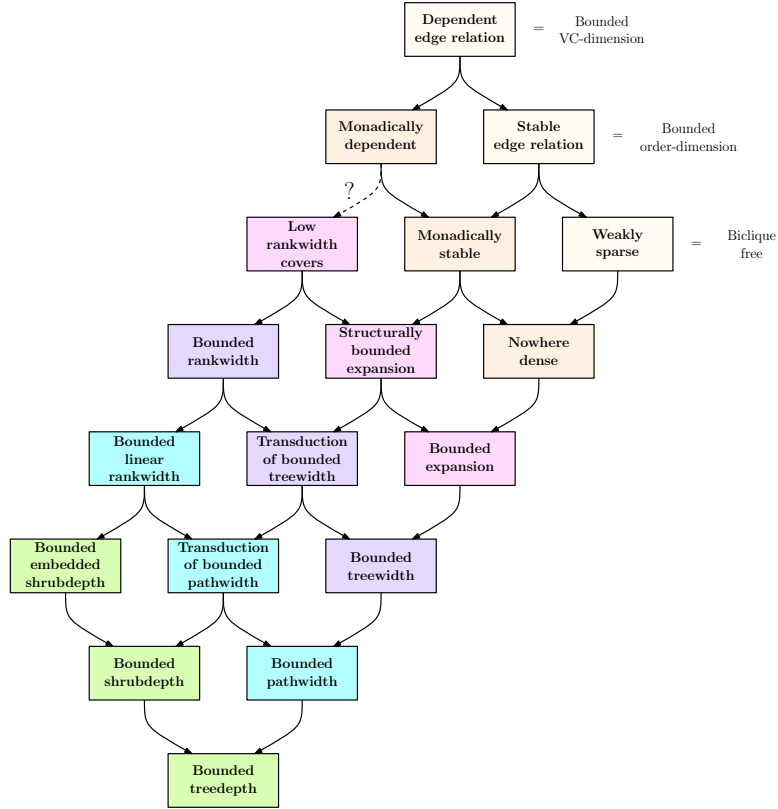


Figure 5: The extended semi-lattice of property inclusions.

every structurally nowhere dense class of graphs is monadically stable, but the precise relationship between these notions remains to be understood. It would be even consistent with our knowledge if the two concepts coincided for classes of graphs. If this was true, it would reveal very strong structural qualities of monadically stable classes of graphs, which could be used in the algorithmic context.

Conjecture 6.1. *A graph class is monadically stable if and only if it is structurally nowhere dense.*

Obviously, besides classes of bounded pathwidth or treewidth, there are multiple other notions of sparsity whose structural analogs could be investigated. For instance, can we characterize *structurally planar* classes, that is, images of the class of planar graphs under transductions? More generally, one may consider images under transductions of classes with forbidden minors or with forbidden topological minors. So far, suitable characterizations have been given for classes with *structurally bounded degree* [16] and with *structurally bounded expansion* [18]. Such characterizations, if efficiently constructive, are very helpful in the design of fixed-parameter algorithms for the FO model-checking problem, as was done in the case of classes with structurally bounded degree [16]. Based on the understanding revealed in [16, 18], we hypothesize that such characterizations may rely on the concept of *covers* (see Section 4). For instance, transductions of classes with bounded expansion are characterized by the existence of such covers (see Theorem 4.2). This motivates the following:

Conjecture 6.2. *Every class with low rankwidth covers is monadically dependent.*

Finally, we recall the conjecture we posed in Section 4.

Conjecture 4.1. *For every structurally bounded expansion class \mathcal{C} , computing a low shrubdepth cover of a graph $G \in \mathcal{C}$ at depth p is fixed parameter tractable when parameterized by p .*

References

- [1] H. Adler and I. Adler. Interpreting nowhere dense graph classes as a classical notion of model theory. *European Journal of Combinatorics*, 36:322–330, 2014.
- [2] P. J. Anderson. Tree-decomposable theories. Master’s thesis, Department of Mathematics and Statistics, Simon Fraser University, 1990.
- [3] D. Angluin. Computational learning theory: survey and selected bibliography. In *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*, pages 351–369, 1992.
- [4] J. T. Baldwin and S. Shelah. Second-order quantifiers and the complexity of theories. *Notre Dame Journal of Formal Logic*, 26(3):229–303, 1985.
- [5] M. Bonamy and M. Pilipczuk. Graphs of bounded cliquewidth are polynomially χ -bounded. *Advances in Combinatorics*, 2020(8). 21pp.
- [6] H. Brönnimann and M. T. Goodrich. Almost optimal set covers in finite VC-dimension. *Discrete & Computational Geometry*, 14(4):463–479, Dec 1995.
- [7] T. Colcombet. A combinatorial theorem for trees. In *Proceedings of the 34th International Colloquium on Automata, Languages and Programming, ICALP 2007*, volume 4596 of *Lecture Notes in Computer Science*, pages 901–912. Springer, 2007.
- [8] B. Courcelle. The monadic second-order logic of graphs. i. recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990.
- [9] B. Courcelle, J. Engelfriet, and G. Rozenberg. Handle-rewriting hypergraph grammars. *Journal of Computer and System Sciences*, 46(2):218–270, 1993.
- [10] B. Courcelle, J. A. Makowsky, and U. Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33(2):125–150, 2000.
- [11] Z. Dvořák. A stronger structure theorem for excluded topological minors. 2012.
- [12] Z. Dvořák. Induced subdivisions and bounded expansion. *European Journal of Combinatorics*, 69:143–148, 2018.
- [13] Z. Dvořák, D. Král, and R. Thomas. Testing first-order properties for subclasses of sparse graphs. *Journal of the ACM (JACM)*, 60(5):1–24, 2013.
- [14] P. Erdős and R. Rado. A combinatorial theorem. *Journal of the London Mathematical Society*, 1(4):249–255, 1950.
- [15] G. Fabiański, M. Pilipczuk, S. Siebertz, and S. Toruńczyk. Progressive algorithms for domination and independence. In *36th International Symposium on Theoretical Aspects of Computer Science, STACS 2019*, volume 126 of *LIPICs*, pages 27:1–27:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.
- [16] J. Gajarský, P. Hliněný, J. Obdržálek, D. Lokshtanov, and M. S. Ramanujan. A new perspective on FO model checking of dense graph classes. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2016*, pages 176–184. ACM, 2016.
- [17] J. Gajarský and D. Král. Recovering sparse graphs. In *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

- [18] J. Gajarský, S. Kreutzer, J. Nešetřil, P. Ossona de Mendez, M. Pilipczuk, S. Siebertz, and S. Toruńczyk. First-order interpretations of bounded expansion classes. *ACM Trans. Comput. Logic*, 21(4):article no. 29, 2020.
- [19] R. Ganian, P. Hliněný, J. Nešetřil, J. Obdržálek, and P. Ossona de Mendez. Shrub-depth: Capturing height of dense graphs. *Logical Methods in Computer Science*, 15(1), 2019. oai:arXiv.org:1707.00359.
- [20] R. Ganian, P. Hliněný, J. Nešetřil, J. Obdržálek, P. Ossona de Mendez, and R. Ramadurai. When trees grow low: Shrubs and fast MSO_1 . In *International Symposium on Mathematical Foundations of Computer Science*, volume 7464 of *Lecture Notes in Computer Science*, pages 419–430. Springer-Verlag, 2012.
- [21] P. A. Golovach, D. Lokshtanov, S. Saurabh, and M. Zehavi. Cliquewidth III: the odd case of graph coloring parameterized by cliquewidth. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 262–273. SIAM, 2018.
- [22] M. Grohe, S. Kreutzer, and S. Siebertz. Deciding first-order properties of nowhere dense graphs. *Journal of the ACM (JACM)*, 64(3):1–32, 2017.
- [23] M. Grohe and D. Marx. Structure theorem and isomorphism test for graphs with excluded topological subgraphs. *SIAM Journal on Computing*, 44(1):114–159, 2015.
- [24] M. Grohe and G. Turán. Learnability and definability in trees and similar structures. *Theory of Computing Systems*, 37(1):193–220, 2004.
- [25] F. Gurski and E. Wanke. The tree-width of clique-width bounded graphs without $K_{n,n}$. In *Proceedings of the 26th International Workshop on Graph-Theoretic Concepts in Computer Science, WG 2000*, volume 1928 of *Lecture Notes in Computer Science*, pages 196–205. Springer, 2000.
- [26] A. Gyárfás. *Problems from the world surrounding perfect graphs*. Number 177. MTA Számítástechnikai és Automatizálási Kutató Intézet, 1985.
- [27] W. Hodges and H. Wilfrid. *Model theory*. Cambridge University Press, 1993.
- [28] Ö. Johansson. Clique-decomposition, NLC-decomposition, and modular decomposition-relationships and results for random graphs. In *Congressus Numerantium*, pages 39–60, 1998.
- [29] O. Kwon, M. Pilipczuk, and S. Siebertz. On low rank-width colorings. *Eur. J. Comb.*, 83, 2020.
- [30] M. Malliaris and S. Shelah. Regularity lemmas for stable graphs. *Transactions of the American Mathematical Society*, 366(3):1551–1585, 2014.
- [31] J. Matoušek. Bounded VC-dimension implies a fractional Helly theorem. *Discrete & Computational Geometry*, 31(2):251–255, 2004.
- [32] J. Nešetřil and P. Ossona de Mendez. Grad and classes with bounded expansion I. decompositions. *European Journal of Combinatorics*, 29(3):760–776, 2008.
- [33] J. Nešetřil and P. Ossona de Mendez. *Sparsity: Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and Combinatorics*. Springer, 2012.
- [34] J. Nešetřil, P. Ossona de Mendez, R. Rabinovich, and S. Siebertz. Linear rankwidth meets stability. In *Proceedings of the 31st ACM-SIAM Symposium on Discrete Algorithms, SODA 2020*, pages 1180–1199, 2020.
- [35] J. Nešetřil, P. Ossona de Mendez, R. Rabinovich, and S. Siebertz. Linear rankwidth meets stability. *European Journal of Combinatorics*, 2020. Special issue dedicated to Xuding Zhu’s 60th birthday (accepted).

- [36] S.-i. Oum and P. D. Seymour. Approximating clique-width and branch-width. *Journal of Combinatorial Theory, Series B*, 96(4):514–528, 2006.
- [37] M. Pilipczuk, M. Pilipczuk, and S. Siebertz. Lecture notes for the course “Sparsity” given at Faculty of Mathematics, Informatics, and Mechanics of the University of Warsaw, Winter Semesters 2017/18 and 2019/20. Available at <https://www.mimuw.edu.pl/mp248287/sparsity2>.
- [38] N. Robertson and P. D. Seymour. Graph Minors. XVI. Excluding a non-planar graph. *Journal of Combinatorial Theory, Series B*, 89(1):43–76, 2003.
- [39] I. Schiermeyer and B. Randerath. Polynomial χ -binding functions and forbidden induced subgraphs: A survey. *Graphs and Combinatorics*, 35:1–31, 2019.
- [40] A. Scott and P. Seymour. A survey of χ -boundedness. 2018.
- [41] E. Wanke. k -NLC graphs and polynomial algorithms. *Discrete Applied Mathematics*, 54(2-3):251–266, 1994.