Automata Approach to Graphs of Bounded Rank-width

Robert Ganian* and Petr Hliněný**

Faculty of Informatics, Masaryk University
Botanická 68a, 602 00 Brno, Czech Republic
ganian@mail.muni.cz, hlineny@fi.muni.cz

Abstract. Rank-width is a rather new structural graph measure introduced by Oum and Seymour in 2003 in order to find an efficiently computable approximation of clique-width of a graph. Being a very nice graph measure indeed, the only serious drawback of rank-width was that it is virtually impossible to use a given rank-decomposition of a graph for running dynamic algorithms on it. We propose a new independent description of rank-decompositions of graphs using labeling parse trees which is, after all, mathematically equivalent to the recent algebraic graph-expression approach to rank-decompositions of Courcelle and Kanté [WG'07]. We then use our labeling parse trees to build a Myhill-Nerode-type formalism for handling restricted classes of graphs of bounded rank-width, and to directly prove that (an already indirectly known result) all graph properties expressible in MSO logic are decidable by finite automata running on the labeling parse trees.

Keywords: graph, parameterized algorithm, rank-width, clique-width, tree automaton, MSO logic.

1 Introduction

Most graph problems are known to be NP-hard in general, and yet a solution to these is needed for practical applications. One common method to provide such a solution is through restricting the input graph to have a certain structure. Often the input graphs are restricted to have bounded tree-width [19] (or branch-width), but a useful weaker structural restriction has been brought by the notion of *clique-width*, defined by Courcelle and Olariu in [7].

Now, many hard graph problems (in particular all those expressible in MSO logic of adjacency graphs) are solvable in polynomial time [6, 10, 16, 12], as long as the input graph has bounded clique-width and is given in the form of the "decomposition for clique-width", called a *k-expression.* A k-expression is an algebraic expression with the following four operations on vertex-labeled graphs using k labels: create a new vertex with label i; take the disjoint union of two labeled graphs; add all edges between vertices of label i and label j; and relabel

* Supported by the Institute for Theoretical Computer Science ITI, project 1M0545.
** Supported by the Czech research grant GAČR 201/08/0308 and project 1M0545.

all vertices with label i to have label j. However, for fixed $k > 3$, it is not known how to find a k-expression of an input graph having clique-width at most k.

Rank-width is another graph complexity measure introduced by Oum and Seymour [18, 8, 17], aiming at providing an $f(k)$-expression of the input graph having clique-width k for some fixed function f in polynomial time. Rank-width is defined (see Section 2) as the branch-width of a so-called *cut-rank* function of graphs. Rank-width turns out to be very useful for algorithms on graphs of bounded clique-width since it can be computed, together with an optimal decomposition, in time $O(n^3)$ on n-vertex graphs of bounded rank-width [15]. Moreover, if rank-width of a graph is k, then its clique-width lies between k and $2^{k+1} - 1$ [18] and a corresponding expression can be constructed from a rank-decomposition of width k.

In view of the previous facts, particularly that clique-width can be up to exponentially larger than rank-width [4], it appears desirable to design efficient algorithms running straight on an optimal rank-decomposition rather than transforming a width-k rank-decomposition into an $f(k)$-expression. Unfortunately, this goal seems practically impossible in a direct way given the rather "strange nature" of a rank-decomposition. Thus one has to look for indirect approaches, say those inspired by a natural geometric link of rank-decompositions of graphs to branch-decompositions of binary matroids [15].

In 2007 Courcelle and Kanté [5] gave an alternative characterization of rank-decompositions of graphs using algebraic terms over multi-coloured graphs. Independently from them, the first author's Master thesis [11] has recently brought the concept of *labeling parse trees* (also called rank-width parse trees) which exactly characterize decompositions of graphs of given rank-width, too. We postpone all the technical definitions till the next section. For now we just note that the latter approach of [11] turns out to be exactly equivalent to the former of [5], though they come from different perspectives.

The aim of this paper is to continue in the labeling parse–tree approach in order to bring a different new view of dynamic algorithms on graphs of bounded rank-width, which is more computer–science oriented (and hence perhaps better understandable among the CS audience) than the algebraic–logic view of Courcelle and Kanté. We make an effort to employ finite tree automata in the task and develop a Myhill–Nerode kind of a characterization of the finite state properties of graphs of bounded rank-width. This viewpoint is not new in other areas, whereas it has been inspired by an analogous handling of bounded tree-width graphs by Abrahamson and Fellows [1] (or [9, Chapter 6]) and of represented matroids of bounded branch-width by the second author [14].

Our characterization then leads to an elementary proof that all MSO expressible graph properties (MS_1, to be precise) are decidable by tree automata over our labeling parse trees, and hence solvable in linear time. We suggest that this new characterization and the related proof techniques may be of independent interest among computer scientists who are working in designing dynamic algorithms for graphs of bounded rank-width (see Section 5 for further discussion).

2 Definitions and Basics

We consider finite simple undirected graphs by default. Here we bring some technical definitions and basic results which are the building blocks of our research.

Branch-width. A set function $f : 2^M \to \mathbb{Z}$ is called *symmetric* if $f(X) = f(M \setminus X)$ for all $X \subseteq M$. A tree is *subcubic* if all its nodes have degree at most 3. For a symmetric function $f : 2^M \to \mathbb{Z}$ on a finite set M, the branch-width of f is defined as follows.

A *branch-decomposition* of f is a pair (T, μ) of a subcubic tree T and a bijective function $\mu : M \to \{t : t \text{ is a leaf of } T\}$. For an edge e of T, the connected components of $T \setminus e$ induce a bipartition (X, Y) of the set of leaves of T. The *width* of an edge e of a branch-decomposition (T, μ) is $f(\mu^{-1}(X))$. The *width* of (T, μ) is the maximum width over all edges of T. The *branch-width* of f is the minimum of the width of all branch-decompositions of f. (If $|M| \leq 1$, then we define the branch-width of f as $f(\emptyset)$.)

A natural application of this definition is the branch-width of a graph, as introduced by Robertson and Seymour [19] along with better known tree-width, and its natural matroidal counterpart. In that case we use $M = E(G)$, and f the connectivity function of G. There is, however, another interesting application of the aforementioned general notions, in which we consider the vertex set $V(G) = M$ of a graph G as the ground set.

Rank-width. For a graph G, let $\boldsymbol{A}_G[U, W]$ be the bipartite adjacency matrix of a bipartition (U, W) of the vertex set $V(G)$ defined over the two-element field GF(2) as follows: the entry $a_{u,w}$, $u \in U$ and $w \in W$, of $\boldsymbol{A}_G[U, W]$ is 1 if and only if uw is an edge of G. The *cut-rank* function $\rho_G(U) = \rho_G(W)$ then equals the rank of $\boldsymbol{A}_G[U, W]$ over GF(2). A *rank-decomposition* and *rank-width* of a graph G is the branch-decomposition and branch-width of the cut-rank function ρ_G of G on $M = V(G)$, respectively.

The main reason for the popularity of rank-width over clique-width is the fact that there are parameterized algorithms for rank-decompositions [18, 15].

Theorem 2.1 (Hliněný and Oum [15]). *For every fixed t there is an $O(n^3)$-time algorithm that, for a given n-vertex graph G, either finds a rank-decomposition of G of width at most t, or confirms that the rank-width of G is more than t.*

Few rank-width examples. Any complete graph of more than one vertex has clearly rank-width 1 since any of its bipartite adjacency matrices consists of all 1s. It is similar with complete bipartite graphs if we split the decomposition along the parts. We illustrate the situation with graph cycles: while C_3 and C_4 have rank-width 1, C_5 and all longer cycles have rank-width equal 2. A rank-decomposition of, say, the cycle C_5 is shown in Fig. 1. Conversely, every subcubic tree with at least 4 leaves has an edge separating at least 2 leaves on each side, and every corresponding bipartition of C_5 gives a matrix of rank ≥ 2.

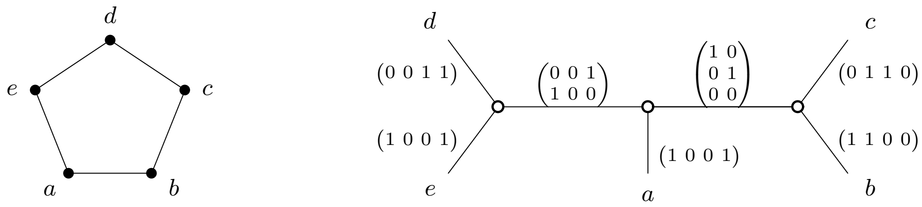

Fig. 1. A rank-decomposition of the graph cycle C_5.

We also mention so-called *distance-hereditary* graphs, i.e. graphs such that the distances in any of their connected induced subgraphs are the same as in the original graph, which have been independently studied, e.g. [3], before. It turns out that distance-hereditary graphs are exactly the graphs of rank-width 1 [17], and this simple fact explains many of their "nice" algorithmic properties.

Labeling parse trees. A (vertex) *t-labeling* of a graph is a mapping $lab : V(G) \to 2^{L_t}$ where $L_t = \{1, 2, \dots, t\}$ is the set of labels (this notion is equivalent to so-called multicoloured graphs of [5]). Having a graph G with an (implicitly) associated t-labeling lab, we refer to the pair G, lab as to a *t-labeled graph* and use notation $\bar{G}$. Notice that each vertex of a t-labeled graph may have zero, one or more labels. So even an unlabeled graph can be considered as t-labeled with no labels, and every t-labeled graph is also t'-labeled for all $t' > t$.

A *t-relabeling* is a mapping $f : L_t \to 2^{L_t}$. For a t-labeled graph $\bar{G} = (G, lab)$ we define $f(\bar{G})$ as the same graph with a vertex t-labeling $lab' = f \circ lab$. Notice that—since the values of lab are subsets of L_t, or vectors from $\mathrm{GF}(2)^t$—the relabeling f in the composition $f \circ lab$ acts as a linear transformation in the vector space $\mathrm{GF}(2)^t$. Informally, f is applied separately to each label in $lab(v)$ and the outcomes are summed up "modulo 2"; such as for $lab(v) = \{1, 2\}$ and $f(1) = \{1, 3, 4\}$, $f(2) = \{1, 2, 3\}$, we get $lab'(v) = \{2, 4\} = \{1, 3, 4\} \vartriangle \{1, 2, 3\}$.

Let $\odot$ be a nullary operator creating a single new graph vertex of label $\{1\}$. For relabelings $f_1, f_2, g : L_t \to 2^{L_t}$ let $\oplus[g \mid f_1, f_2]$ be a binary operator over pairs of t-labeled graphs $\bar{G}_1 = (G_1, lab^1)$ and $\bar{G}_2 = (G_2, lab^2)$ defined as follows:

$$(G_1, lab^1) \oplus[g \mid f_1, f_2] \ (G_2, lab^2) \ = \ (H, lab)$$

where the graph H is constructed from the disjoint union $G_1 \dot\cup G_2$ by adding all edges uw, $u \in V(G_1)$ and $w \in V(G_2)$ such that $|lab^1(u) \cap g \circ lab^2(w)|$ is odd, and with the new labeling $lab(v) = f_i \circ lab^i(v)$ for $v \in V(G_i)$, $i = 1, 2$.

A *labeling parse tree* T, see [11, Definition 6.11], is a finite rooted ordered subcubic tree (with the root degree at most 2) such that

– all leaves of T contain the $\odot$ operator, and
– all internal nodes of T contain one of the $\oplus[g \mid f_1, f_2]$ operators.

A parse tree T then *generates* (parses) the graph G which is obtained by successive leaves-to-root applications of the operators in the nodes of T. See Fig. 2.

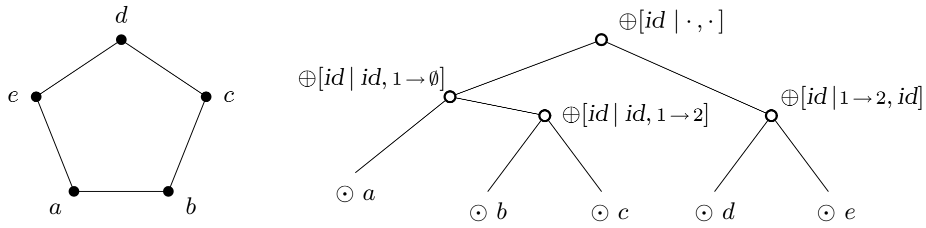

Fig. 2. An example of a labeling parse tree which generates a 2-labeled cycle C_5, with symbolic operators at the nodes (*id* denotes the relabeling preserving all labels).

The following substantial Theorem 2.2 is actually equivalent to [5, Theorem 3.4]. Its independent detailed proof can be found in the first author's Master thesis [11, Chapter 6] (the time complexity bound being implicit there).

Theorem 2.2 (Rank-width parsing theorem [11]). *A graph G has rank-width at most t if and only if (some labeling of) G can be generated by a labeling parse tree using t labels. Furthermore, an optimal rank-decomposition of G can be transformed into a labeling parse tree with t labels in time $O(n^2)$.*

We add a short note that time complexity $O(n^2)$ can be considered "linear" in this case since the size of the graph G can be of order up to n^2. We suggest that this complexity can be improved even to $O(|E(G)|)$ if one carefully reconsiders all the technical details, but that would not be useful in our context in which we use Theorem 2.2 together with Theorem 2.1 to construct an optimal labeling parse tree of a given graph G in parameterized $O(n^3)$ time.

3 Regularity Theorem for Rank-width

The core new contribution of our paper lies in developing a mathematical formalism for easy handling of graph properties which are efficiently solvable on graphs of bounded rank-width. Our formalism is closely tied with the classical Myhill–Nerode regularity theorem in automata theory. As we have already noted above, we are inspired by analogous formalisms used in [1] (graphs of bounded tree-width) and in [14] (matroids of bounded branch-width).

Recalling the notation of labeling parse trees, we shortly write $\oplus[g]$ for $\oplus[g \,|\, \emptyset, \emptyset]$ where $\emptyset$ stands for the relabeling $L \to \{\emptyset\}$ "forgetting" all vertex labels. Notice that the binary operation $\oplus[g]$ which creates an unlabeled graph from two labeled graphs is not commutative, but its operands can be exchanged together with a suitable modification of g. The role of a specific relabeling g in $\oplus[g]$ is rather technical after all, as the next immediate claim says. Let *id* be the relabeling preserving all labels.

Proposition 3.1. *Let G_1, G_2 be t-labeled graphs generated by labeling parse trees T_1, T_2, and $g : L_t \to 2^{L_t}$ be any relabeling. Then there is a tree T_2^g parsing a t-labeled graph G_2^g (actually unlabeled-equal to G_2) such that*

$$G_1 \oplus[g]\; G_2 = G_1 \oplus[id]\; G_2^g \,.$$

Canonical equivalence. Let Π_t denote the finite set (alphabet) of operators of labeling parse trees with t labels, and let subsequently $P_t \subseteq \Pi_t^{**}$ be the class (language) of all valid labeling parse trees with t labels. If $\mathcal{R}_t$ denotes the class of all unlabeled graphs of rank-width at most t and $\overline{\mathcal{R}}_t$ is the class of all t-labeled graphs parsed by the trees from P_t, then (Theorem 2.2) $G \in \mathcal{R}_t$ if and only if $\bar{G} \in \overline{\mathcal{R}}_t$ for some t-labeling $\bar{G}$ of G.

Let $\mathcal{D}$ be any class of graphs, and $\mathcal{D}_t = \mathcal{D} \cap \mathcal{R}_t$. In analogy to classical theory of regular languages we define a *canonical equivalence* of $\mathcal{D}_t$, denoted by $\approx_{\mathcal{D},t}$, as follows: $\bar{G}_1 \approx_{\mathcal{D},t} \bar{G}_2$ for any $\bar{G}_1, \bar{G}_2 \in \overline{\mathcal{R}}_t$ if and only if, for all $\bar{H} \in \overline{\mathcal{R}}_t$,

$$\bar{G}_1 \oplus [id]\ \bar{H} \in \mathcal{D}_t \iff \bar{G}_2 \oplus [id]\ \bar{H} \in \mathcal{D}_t\,.$$

In informal words, the classes of $\approx_{\mathcal{D},t}$ "capture" all information we need to know about a t-labeled subgraph $\bar{G} \in \overline{\mathcal{R}}_t$ to decide membership in $\mathcal{D}$ further on in our parse tree processing.

The previous informal finding can be formalized as follows:

Theorem 3.2 (Rank-width regularity theorem). *Let $t \geq 1$, $\mathcal{D}$ be a graph class, and $\mathcal{D}_t = \mathcal{D} \cap \mathcal{R}_t$. The collection of all those labeling parse trees which generate the members of $\mathcal{D}_t$ is accepted by a finite tree automaton if, and only if, the canonical equivalence $\approx_{\mathcal{D},t}$ of $\mathcal{D}_t$ over $\overline{\mathcal{R}}_t$ is of finite index.*

Sketch of proof. A detailed proof of this statement is contained in [11, Chapter 7]. We only sketch its main ideas due to space restrictions here.

Our starting point is the classical Myhill–Nerode theorem for tree automata. Let Σ^{**} denote the set of all rooted binary trees over a finite alphabet Σ. For a language $\lambda \subseteq \Sigma^{**}$ we can define a congruence $\sim_\lambda$ such that $T_1 \sim_\lambda T_2$ for $T_1, T_2 \in \Sigma^{**}$ if, and only if, $T_1 \diamond_x U \in \lambda \iff T_2 \diamond_x U \in \lambda$ where U runs over all special rooted binary trees over Σ with one distinguished leaf node x, and $T_i \diamond_x U$ results from U by replacing the leaf x with the subtree T_i. Then λ is accepted by a finite tree automaton if and only if $\sim_\lambda$ has finite index.

In our case $\Sigma = \Pi_t$, and λ are the labeling parse trees of the members of $\mathcal{D}_t$. So, to prove our theorem it is enough to show that $\approx_{\mathcal{D},t}$ has infinite index if and only if $\sim_\lambda$ has infinite index.

Suppose the former holds, i.e. there are infinitely many $\bar{G}_k \in \overline{\mathcal{R}}_t$, $k = 1, 2, \ldots$, such that for all indices $i \neq j$ there exists $\bar{H}_{i,j} \in \overline{\mathcal{R}}_t$ for which $\bar{G}_i \oplus [id]\ \bar{H}_{i,j} \in \mathcal{D}_t$ but $\bar{G}_j \oplus [id]\ \bar{H}_{i,j} \notin \mathcal{D}_t$, or vice versa. Let S_k be a labeling parse tree of $\bar{G}_k$, and $Q_{i,j}$ that of $\bar{H}_{i,j}$. We define a new parse tree $U_{i,j}$ such that the root operator is $\oplus[id \mid \emptyset, \emptyset]$, its left son is the distinguished leaf x, and its right subtree is $Q_{i,j}$. Hence the special trees $U_{i,j}$ witness that all the parse trees S_k, $k = 1, 2, \ldots$ belong to distinct classes of $\sim_\lambda$.

Conversely, suppose that the latter holds. So there are infinitely many trees $S_k \in \Pi_t^{**}$, $k = 1, 2, \ldots$, such that for each pair of indices $i \neq j$ there exists $U_{i,j}$ as above for which $S_i \diamond_x U_{i,j} \in \lambda$ but $S_j \diamond_x U_{i,j} \notin \lambda$, or vice versa. We may assume without loss of generality that $S_k \in P_t$ are valid labeling parse trees for all k. Let $\bar{G}_k$ be the graphs parsed by S_k. Using technical [11, Lemma 7.3] and Proposition 3.1, we deduce that there exist graphs $\bar{H}_{i,j}$ such that

– the graph parsed by $S_i \diamond_x U_{i,j}$ is equal up to labeling to $\bar{G}_i \oplus[id]\ \bar{H}_{i,j} \in \mathcal{D}_t$,
– and the graph parsed by $S_j \diamond_x U_{i,j}$ equals up to labeling $\bar{G}_j \oplus[id]\ \bar{H}_{i,j} \notin \mathcal{D}_t$.

This assertion certifies that the graphs $\bar{G}_k$ indeed belong to distinct classes of our canonical equivalence $\approx_{\mathcal{D},t}$. ∎

Remark 3.3. Notice that the arguments used in our proof of Theorem 3.2 *do not* straightforwardly translate from rank-width (and labeling parse trees) to clique-width (and its k-expressions). Quite the opposite, the "only if" direction of this theorem seems not at all provable in the above way since one cannot freely choose the "root" of a k-expression (cf. [11, Lemma 7.3]). We consider that another small reason to favor rank-width over clique-width in CS applications.

3-colourability example. We briefly demonstrate the use of Theorem 3.2 on graph 3-colourability which is a well-known NP-complete problem. Let $\mathcal{C}$ denote the class of all simple 3-colourable graphs. To construct a tree automaton accepting the labeling parse trees of the members of $\mathcal{C}_t = \mathcal{C} \cap \mathcal{R}_t$, it is enough to identify the classes of the canonical equivalence $\approx_{\mathcal{C},t}$. We actually give the finitely many classes $X_0,\ X_1, X_2, \dots$ of the following refinement of $\approx_{\mathcal{C},t}$:

– $X_0 = \{G :\ G \text{ is not 3-colourable}\}$.
– Otherwise, for any t-labeled graph G with a proper 3-colouring χ, we define a vector $c(G,\chi) = (c_\ell :\ \ell \in 2^{L_t})$ where $c_\ell \subseteq \{1,2,3\}$ is the set of χ-colours occurring in the vertices of G labeled by ℓ.
  $X_1, X_2, \dots, X_{h(t)}$ are then the equivalence classes of $\sim$, where over t-labeled graphs $G_1 \sim G_2$ if and only if it holds $\{c(G_1,\chi) :\ \chi \text{ is a proper 3-colouring of } G_1\} = \{c(G_2,\chi) :\ \chi \text{ is a proper 3-colouring of } G_2\}$.

4 Regularity and MSO Definable Properties

From a logic point of view, we consider a graph as a relational structure on the ground set V, with one binary predicate $edge(u,v)$. When the language of MSO logic is applied to such a graph adjacency structure, one gets a descriptional language over graphs commonly abbreviated as MS_1. For an illustration we show an MS_1 expression of the 3-colourability property of a graph:

$$\exists V_1, V_2, V_3 \big[\quad \forall v\ (v \in V_1 \vee v \in V_2 \vee v \in V_3)\ \wedge \\ \textstyle\bigwedge_{i=1,2,3} \forall v, w\ (v \notin V_i \vee w \notin V_i \vee \neg\, edge(v,w)) \big]$$

It is also common to consider the "counting" version of MSO logic which moreover has predicates $mod_{p,q}(X)$ stating that $|X| \mod p = q$.

To avoid possible confusion we remark ahead that there is a stronger descriptional language MS_2 of graphs which allows to quantify also over graph edges and their sets, and which is related to graphs of bounded tree-width. There are MS_2 expressible graph properties, e.g. Hamiltonicity, which are not expressible in MS_1, whilst MS_2 properties cannot be efficiently handled in general on graphs of bounded rank-width.

In [6] Courcelle, Makowsky, and Rotics proved that all MS_1 definable graph properties are solvable in linear time, in fact by a tree automaton, running on a given k-expression (k fixed) of the graph. Their indirect proof used MSO interpretation (transduction) of the graphs generated by k-expressions into labeled binary trees. Since a graph class has bounded clique-width if and only if it has bounded rank-width, the results of [6] carry over to graphs of bounded rank-width (with a possible exponential jump in the width parameter).

We, on the other hand, favor the independent direct combinatorial approach to these problems, paralleling [1, 14]:

Theorem 4.1. *Let $t \geq 1$. If $\mathcal{D}$ is a graph class definable in the MS_1 language, then the collection of all those labeling parse trees which generate the members of $\mathcal{D}_t = \mathcal{D} \cap \mathcal{R}_t$ is accepted by a finite tree automaton.*

To prove this statement, in view of Theorem 3.2, it is enough to prove that the associated canonical equivalence $\approx_{\mathcal{D},t}$ is of finite index. However, the latter claim needs a generalization in order to use mathematical induction on the structure of an MS_1 sentence ϕ describing $\mathcal{D}$. This generalization of $\approx_{\mathcal{D},t}$ to $\approx^{\circ}_{\phi,t}$ lies in allowing formulas ϕ with free variables.

Extended canonical equivalence of ϕ. Let $Free(\phi) = Fr(\phi) \cup FR(\phi)$ be the partition of the free variables into those $Fr = Fr(\phi)$ for vertices and those $FR = FR(\phi)$ for vertex sets. We define a *partial equipment signature* of ϕ as a triple $\sigma = (Fr, FR, q)$ where $q : Fr \to \{0, 1\}$. A t-labeled graph G is σ-*partially equipped* if it has distinguished vertices and vertex sets assigned as interpretations of the free variables in σ. Formally, for each $X \in FR$ there is a distinguished subset $S_X \subseteq V(G)$, and for each $x \in Fr$ such that $q(x) = 0$ there is a distinguished vertex $v_x \in V(G)$. Nothing is assigned to variables $x \in Fr$ such that $q(x) = 1$. For σ we define a *complemented* partial equipment signature $\sigma^- = (Fr, FR, q')$ where $q'(x) = 1 - q(x)$ for all $x \in Fr$.

See that if $\bar{H}_1$ is σ-partially equipped and $\bar{H}_2$ is σ^--partially equipped, then $H = \bar{H}_1 \oplus[g]\ \bar{H}_2$ has a full and consistent interpretation for all the free variables of ϕ (hence this H is a logic model of ϕ). So, we can define equivalence $\approx^{\sigma}_{\phi,t}$ over all t-labeled σ-partially equipped graphs as follows: $\bar{G}_1 \approx^{\sigma}_{\phi,t} \bar{G}_2$ if and only if the following

$$(\bar{G}_1 \oplus[id]\ \bar{H}) \models \phi \iff (\bar{G}_2 \oplus[id]\ \bar{H}) \models \phi$$

holds for all t-labeled σ^--partially equipped graphs $\bar{H}$.

Here we have extended the meaning of $\approx^{\sigma}_{\phi,t}$ in two directions. Firstly, by allowing free variables in ϕ we enlarge the studied universe to partially equipped graphs. Secondly, the universe is further enlarged by allowing all t-labeled underlying graphs – not only those from $\overline{\mathcal{R}}_t$. Even in this stronger variant we can prove the following key statement which also concludes above Theorem 4.1:

Theorem 4.2. *Let $t \geq 1$ be fixed. Suppose ϕ is a formula in the language MS_1, and σ is a partial equipment signature for ϕ. Then $\approx^{\sigma}_{\phi,t}$ has finite index on the universe of t-labeled σ-partially equipped graphs.*

Proof. We retain the notation introduced above. The induction base is to prove the statement for the atomic formulas in MS_1: $\phi \equiv (v \in W)$, $(v = w)$, $mod_{p,q}(W)$, or $edge(u, v)$. The first three are all rather trivial cases which we skip here, and we focus on the last predicate $edge(u, v)$ (since this one actually "defines" the graph we study).

(4.3) Suppose $\phi \equiv edge(u, v)$. Then the index of $\approx^{\sigma}_{\phi,t}$ is one if $q(u) = q(v) = 1$, two if $q(u) = q(v) = 0$, and 2^t if $q(u) = 0$ and $q(v) = 1$ or vice versa.

In the first case both vertices u, v with a possible edge uv are interpreted in the right-hand graph $\bar{H}$, and hence no matter what $\bar{G}_1$ or $\bar{G}_2$ are, they become equivalent in $\approx^{\sigma}_{\phi,t}$. In the second case both vertices u, v are interpreted in the left-hand graphs $\bar{G}_i$, and hence there are exactly two classes formed by those graphs having and those not having u adjacent to v. It is the third case which interests us: Recalling the definition of our summation operator $\oplus[id]$, we see that all information needed to decide whether some u in the left-hand graph is adjacent to a specific v in the right-hand graph is encoded in the labeling of u, and hence the 2^t possibilities there.

For the inductive step, we consider that a formula ϕ is created from shorter formula(s) in one of the following ways: $\phi \equiv \neg\psi$, $\psi \wedge \eta$, $\exists v\, \psi(v)$, or $\exists W\, \psi(W)$, where $v \in Fr(\psi)$ or $W \in FR(\psi)$ in the latter cases. One may easily express the $\vee$ or $\forall$ symbols using these. The arguments we are going to give in the rest of this proof are not novel, but similar to those used in [1] and merely a translation of the arguments used in [14, Lemma 6.2].

We assume by induction that $\approx^{\pi}_{\psi,t}$ ($\approx^{\rho}_{\eta,t}$) has finite index, where the signature π (ρ) is inherited from σ for ψ (for η, see below the case-by-case details). The first case of $\phi \equiv \neg\psi$ is quite easy to resolve — the equivalence $\approx^{\pi}_{\psi,t}$ is the same as $\approx^{\sigma}_{\phi,t}$. We look at the second case.

(4.4) Suppose $\phi \equiv \psi \wedge \eta$, and let π, ρ denote the restrictions of signature σ to $Free(\psi)$, $Free(\eta)$, respectively. If $\approx^{\pi}_{\psi,t}$ has index p and $\approx^{\rho}_{\eta,t}$ has index r, then $\approx^{\sigma}_{\phi,t}$ has index at most $p \cdot r$.

Consider an arbitrary pair of t-labeled σ-partially equipped graphs $\bar{G}_1 \not\approx^{\sigma}_{\phi,t} \bar{G}_2$, and an associated σ^--partially equipped graph $\bar{H}$ such that $(\bar{G}_1 \oplus[id]\, \bar{H}) \models \phi$ but $(\bar{G}_2 \oplus[id]\, \bar{H}) \not\models \phi$. Then it has to be $(\bar{G}_1 \oplus[id]\, \bar{H}) \models \psi$ (or $\models \eta$) but $(\bar{G}_2 \oplus[id]\, \bar{H}) \not\models \psi$ (or $\not\models \eta$, resp.). Hence it immediately holds that $\bar{G}_1 \not\approx^{\pi}_{\psi,t} \bar{G}_2$ or $\bar{G}_1 \not\approx^{\rho}_{\eta,t} \bar{G}_2$ with the restricted equipments, and so the equivalence classes of $\approx^{\sigma}_{\phi,t}$ are suitable unions of the classes of the "intersection" $\approx^{\pi}_{\psi,t} \cap \approx^{\rho}_{\eta,t}$.

The third case of $\exists v\, \psi(v)$ is technically more complicated, and so we first deal with the similar but easier fourth case of $\exists W\, \psi(W)$.

(4.5) Suppose $\phi \equiv \exists W\, \psi(W)$, and let the signature $\pi = (Fr, FR \cup \{W\}, q)$. If $\approx^{\pi}_{\psi,t}$ has index p, then $\approx^{\sigma}_{\phi,t}$ has index at most $2^p - 1$.

Again consider an arbitrary pair of t-labeled σ-partially equipped graphs $\bar{G}_1 \not\approx^{\sigma}_{\phi,t} \bar{G}_2$, and $\bar{H}$ such that $(\bar{G}_1 \oplus[id]\, \bar{H}) \models \phi$ but $(\bar{G}_2 \oplus[id]\, \bar{H}) \not\models \phi$. We shortly write $\bar{G}[W = S]$ for the π-partially equipped graph obtained from σ-partially equipped $\bar{G}$ by interpreting the variable W as $S \subseteq V(\bar{G})$. Then our assumption about $\bar{G}_1, \bar{G}_2$ means there exist $S_W \subseteq V(\bar{G}_1)$ and $S'_W \subseteq V(\bar{H})$ such that $\big(\bar{G}_1[W = S_W] \oplus [id]\ \bar{H}[W = S'_W]\big) \models \psi$, whilst $\big(\bar{G}_2[W = T_W] \oplus [id]\ \bar{H}[W = S'_W]\big) \not\models \psi$ for all $T_W \subseteq V(\bar{G}_2)$. Hence $\bar{G}_1[W = S_W] \not\approx^{\pi}_{\psi,t} \bar{G}_2[W = T_W]$.

We now, in search for a contradiction, look at the problem from the other side. Let the equivalence classes of $\approx^{\pi}_{\psi,t}$ over t-labeled π-partially equipped graphs be $\mathcal{C}^1, \mathcal{C}^2, \dots, \mathcal{C}^p$. For a σ-partially equipped graph $\bar{G}$ we define a nonempty set $Ix(\bar{G}) \subseteq \{1, 2, \dots, p\}$ as follows: $i \in Ix(\bar{G})$ if and only if $\bar{G}[W = S] \in \mathcal{C}^i$ for some $S \subseteq V(\bar{G})$. If there were 2^p pairwise incomparable σ-partially equipped graphs in the relation $\approx^{\sigma}_{\phi,t}$, then some two of them, say $\bar{G}_1 \not\approx^{\sigma}_{\phi,t} \bar{G}_2$, would receive $Ix(\bar{G}_1) = Ix(\bar{G}_2)$ by the pigeon-hole principle. However, from the argument of the previous paragraph — $\bar{G}_1[W = S_W] \not\approx^{\pi}_{\psi,t} \bar{G}_2[W = T_W]$ for some $S_W \subseteq V(\bar{G}_1)$ and all $T_W \subseteq V(\bar{G}_2)$, we conclude that $j \in Ix(\bar{G}_1) \setminus Ix(\bar{G}_2)$ where j is such that $\bar{G}_1[W = S_W] \in \mathcal{C}^j$. This contradiction proves (4.5).

(4.6) Suppose $\phi \equiv \exists v\, \psi(v)$, and let signatures $\pi = (Fr \cup \{v\}, FR, q_1)$ and $\rho = (Fr \cup \{v\}, FR, q_2)$ where $q_1(v) = 0$ and $q_2(v) = 1$. If $\approx^{\pi}_{\psi,t}$ has index p and $\approx^{\rho}_{\psi,t}$ has index r, then $\approx^{\sigma}_{\phi,t}$ has index at most $2^p \cdot r + 1 - r$.

Notice that a ρ-partial equipment of $\bar{G}$ does not interpret the variable v in $V(G)$, and so σ-partially equipped graph $\bar{G}$ may be viewed also as ρ-partially equipped. Take an arbitrary pair of nonempty t-labeled σ-partially equipped graphs $\bar{G}_1 \not\approx^{\sigma}_{\phi,t} \bar{G}_2$, and $\bar{H}$ such that $(\bar{G}_1 \oplus [id]\ \bar{H}) \models \phi$ but $(\bar{G}_2 \oplus [id]\ \bar{H}) \not\models \phi$. Let $x_v \in V(\bar{G}_1) \cup V(\bar{H})$ be an interpretation of the variable v that satisfies ψ over $\bar{G}_1 \oplus [id]\ \bar{H}$. In particular, ψ is false over $\bar{G}_2 \oplus [id]\ \bar{H}$ here. If $x_v \in V(\bar{H})$, then immediately $\bar{G}_1 \not\approx^{\rho}_{\psi,t} \bar{G}_2$. Otherwise, $x_v \in V(\bar{G}_1)$ and we are in a situation analogous to the first paragraph of (4.5): $\big(\bar{G}_1[v = x_v] \oplus [id]\ \bar{H}\big) \models \psi$, whilst $\big(\bar{G}_2[v = y_v] \oplus [id]\ \bar{H}\big) \not\models \psi$ for all $y_v \in V(\bar{G}_2)$.

Again, in search for a contradiction, we look at the problem from the other side. If there are $2^p r + 2 - r$ pairwise incomparable σ-partially equipped graphs with respect to $\approx^{\sigma}_{\phi,t}$, then at least $2^p r + 1 - r = (2^p - 1)r + 1$ of those graphs are nonempty, and out of them at least 2^p belong to the same equivalence class of $\approx^{\rho}_{\psi,t}$. Let their set be denoted by $\mathcal{G}$ (Hence for each pair in $\mathcal{G}$, the latter conclusion of the previous paragraph applies). Considering the equivalence classes $\mathcal{C}^1, \mathcal{C}^2, \dots, \mathcal{C}^p$ of $\approx^{\pi}_{\psi,t}$, we again (as in 4.5) define a nonempty set $Ix(\bar{G}) \subseteq \{1, 2, \dots, p\}$, for σ-partially equipped $\bar{G}$, by $i \in Ix(\bar{G})$ if and only if $\bar{G}[v = y] \in \mathcal{C}^i$ for some $y \in V(\bar{G})$. Then some pair, say $\bar{G}_1, \bar{G}_2 \in \mathcal{G}$, must satisfy $Ix(\bar{G}_1) = Ix(\bar{G}_2)$ by the pigeon-hole principle. However, that analogously contradicts the latter conclusion of the previous paragraph.

This contradiction proves (4.6), and thus the whole theorem. ∎

5 Concluding Notes

As already mentioned in the introduction, the driving force of our research is to provide a framework for easy design of efficient parameterized algorithms running on a bounded-width rank-decomposition of a graph. In this sense we have provided two directly applicable results in Theorems 2.2 and 4.1. Unfortunately, applicability of Theorem 4.1 is limited to pure decision problems (like 3-colourability), but many practical problems are formulated as optimization ones. (The usual way of transforming optimization problems into decision ones does not work here since MS_1 logic cannot handle arbitrary numbers.)

Nevertheless, there is a known solution. Arnborg, Lagergren, and Seese [2] (while studying graphs of bounded tree-width), and later Courcelle, Makowsky, and Rotics [6] (for graphs of bounded clique-width), specifically extended the expressive power of MSO logic to define so-called *Lin*EMSO optimization problems, and consequently shown existence of linear-time algorithms in the respective cases. Briefly saying, *Lin*EMSO problems allow, in addition to ordinary MSO expressions, to optimize over and compare between linear evaluation terms.

We can achieve an analogous solution in our framework directly using Theorem 4.2. The basic idea is that, in a dynamic processing of the input parse tree, we can keep track only of suitable "optimal" representatives of the possible interpretations of the free variables in ϕ, per each class of the extended canonical equivalence $\approx^{\sigma}_{\phi,t}$. We illustrate this idea with the next simple example.

Dominating set example. This problem asks for a subset $X \subseteq V(G)$ of the least cardinality such that each vertex not in X is adjacent to some in X. Since it is not a decision question, we cannot hope in a direct application of Theorem 4.1. We, however, can write in MS_1

$$\delta(X) \;\equiv\; \forall v \exists w \left[v \in X \vee \big(w \in X \wedge \mathit{edge}(v,w) \big) \right]$$

stating that X is a dominating set in G. And now Theorem 4.2 can be applied.

Let G be a graph of rank-width t, and T its labeling parse tree. We denote by T_x the subtree below a node x of T, and by G_x the subgraph of G parsed by T_x. For any $D \subseteq V(G_x)$, the t-labeled partially equipped graph G_x with interpretation $X = D$ falls into one of the finitely many classes of $\approx^{\sigma}_{\delta,t}$ (where $\sigma = (\emptyset, \{X\}, \emptyset)$). A dynamic algorithm for the dominating set problem has to remember just one representative interpretation $X = D_i$ of the least cardinality from the i-th class of $\approx^{\sigma}_{\delta,t}$, and with knowledge of the associated tree automaton (Theorem 3.2) this information can easily be processed from leaves of T to the root in total linear time.

Non-FPT algorithms for bounded widths. Lastly we note the following interesting phenomenon: for some problems on graphs of bounded width parameters, there are known algorithms which run faster than in general case, but they are not fixed parameter tractable. Among those we mention a (pseudo)polynomial algorithm for the chromatic number of graphs of bounded clique-width [16], or a subexponential algorithm for the Tutte polynomial of graphs of bounded clique-width [13]. Finite automata clearly cannot be applied there. Though, it would be interesting to extend the framework of Theorem 4.2 to also cover the aforementioned situation.

References

1. Abrahamson, K.A. and Fellows, M.R.: Finite Automata, Bounded Treewidth, and Well-Quasiordering. In: Graph Structure Theory, Contemporary Mathematics 147. American Mathematical Society (1993) 539–564
2. Arnborg, S., Lagergren, J., and Seese, D.: Problems easy for Tree-decomposible Graphs. Proc. 15th Colloq. Automata, Languages and Programming. Volume 317 of Lecture Notes in Comput. Sci. Springer, Berlin (1988) 38–51
3. Bandelt, H.-J. and Mulder, H.M.: Distance-hereditary graphs. J. Combin. Theory Ser. B **41**(2) (1986) 182–208
4. Corneil, D.G., Rotics, U.: On the relationship between cliquewidth and treewidth. SIAM J. Comput. **34**(4) (2005) 825–847
5. Courcelle, B. and Kanté, M.M.: Graph Operations Characterizing Rank-Width and Balanced Graph Expressions. In: Graph-theoretic concepts in computer science, Volume 4769 of Lecture Notes in Comput. Sci., Berlin, Springer (2007) 66–75
6. Courcelle, B., Makowsky, J.A., and Rotics, U.: Linear time solvable optimization problems on graphs of bounded clique-width. Theory Comput. Syst. **33**(2) (2000) 125–150
7. Courcelle, B. and Olariu, S.: Upper bounds to the clique width of graphs. Discrete Appl. Math. **101**(1-3) (2000) 77–114
8. Courcelle, B. and Oum, S.: Vertex-minors, monadic second-order logic, and a conjecture by Seese. J. Combin. Theory Ser. B **97**(1) (2007) 91–126
9. Downey, R.G. and Fellows, M.R.: *Parameterized complexity*. Monographs in Computer Science. Springer-Verlag, New York, 1999.
10. Espelage, W., Gurski, F., and Wanke, E.: How to solve NP-hard graph problems on clique-width bounded graphs in polynomial time. In: Graph-theoretic concepts in computer science. Volume 2204 of Lecture Notes in Comput. Sci., Berlin, Springer (2001) 117–128
11. Ganian, R.: Automata formalization for graphs of bounded rank-width. Master thesis. Faculty of Informatics of the Masaryk University, Brno, Czech republic (2008)
12. Gerber, M.U. and Kobler, D.: Algorithms for vertex-partitioning problems on graphs with fixed clique-width. Theoret. Comput. Sci. **299**(1-3) (2003) 719–734
13. Gimenez, O., Hliněný, P. and Noy, M.: Computing the Tutte Polynomial on Graphs of Bounded Clique-Width. In: Graph-theoretic concepts in computer science. Volume 3787 of Lecture Notes in Comput. Sci., Berlin, Springer (2005) 59–68
14. Hliněný, P.: Branch-width, parse trees, and monadic second-order logic for matroids. J. Combin. Theory Ser. B **96**(3) (2006) 325–351
15. Hliněný, P. and Oum, S.: Finding Branch-decomposition and Rank-decomposition. SIAM J. Comput. (2008) to appear
16. Kobler, D. and Rotics, U.: Edge dominating set and colorings on graphs with fixed clique-width. Discrete Appl. Math. **126**(2-3) (2003) 197–221
17. Oum, S.: Rank-width and vertex-minors. J. Combin. Theory Ser. B **95**(1) (2005) 79–100
18. Oum, S. and Seymour, P.: Approximating clique-width and branch-width. J. Combin. Theory Ser. B **96**(4) (2006) 514–528
19. Robertson, N. and Seymour, P.: Graph minors. X. Obstructions to tree-decomposition. J. Combin. Theory Ser. B **52**(2) (1991) 153–190