Monadic Second Order Finite Satisfiability and Unbounded Tree-Width*

Tomer Kotek[1], Helmut Veith[2], and Florian Zuleger[3]

1 TU Vienna, Austria
2 TU Vienna, Austria
3 TU Vienna, Austria

Abstract

The finite satisfiability problem of monadic second order logic is decidable only on classes of structures of bounded tree-width by the classic result of Seese [25]. We prove that the following problem is decidable:

Input: (i) A monadic second order logic sentence α, and (ii) a sentence β in the two-variable fragment of first order logic extended with counting quantifiers. The vocabularies of α and β may intersect.
Output: Is there a finite structure which satisfies $\alpha \wedge \beta$ such that the restriction of the structure to the vocabulary of α has bounded tree-width? (The tree-width of the desired structure is not bounded.)

As a consequence, we prove the decidability of the satisfiability problem by a finite structure of bounded tree-width of a logic $\mathrm{MSO}^{\exists\mathrm{card}}$ extending monadic second order logic with linear cardinality constraints of the form $|X_1| + \cdots + |X_r| < |Y_1| + \cdots + |Y_s|$ on the variables X_i, Y_j of the outer-most quantifier block. We prove the decidability of a similar extension of WS1S.

1998 ACM Subject Classification F.4.1 Mathematical Logic

Keywords and phrases Monadic Second Order Logic MSO, Two variable Fragment with Counting C2, Finite decidability, Unbounded Tree-width, WS1S with Cardinality Constraints

Digital Object Identifier 10.4230/LIPIcs.CSL.2016.13

1 Introduction

Monadic second order logic (MSO) is among the most expressive logics with good algorithmic properties. It has found countless applications in computer science in diverse areas ranging from verification and automata theory [13, 19, 26] to combinatorics [16, 18], and parameterized complexity theory [8, 6].

The power of MSO is most visible over graphs of bounded tree-width, and with second order quantifiers ranging over sets of edges[1]: (1) Courcelle's famous theorem shows that MSO model checking is decidable over graphs of bounded tree-width in linear time [5, 1]. (2) Finite satisfiability by graphs of bounded tree-width is decidable [5] (with non-elementary complexity) – thus contrasting Trakhtenbrot's undecidability result of first order logic. (3) Seese proved [25] that for each class $\mathcal{K}$ of graphs with *unbounded* tree-width, finite satisfiability of MSO by graphs in $\mathcal{K}$ is undecidable. Together, (2) and (3) give a fairly clear

* The tragic death of Helmut Veith prevented him from approving the final version. All faults and inaccuracies belong to his co-authors.
[1] The logic we denote by MSO is denoted MS_2 by Courcelle and Engelfriet [6].

25th EACSL Annual Conference on Computer Science Logic (CSL 2016).
Editors: Jean-Marc Talbot and Laurent Regnier; Article No. 13; pp. 13:1–13:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

picture of the decidability of finite satisfiability of MSO. It appeared that (3) gives a natural limit for decidability of MSO on graph classes. For instance, finite satisfiability on planar graphs is undecidable because their tree-width is unbounded.

While Courcelle and Seese circumvent Trakhtenbrot's undecidability result by *restricting the classes of graphs (or relational structures)*, several other research communities have studied syntactic restrictions of first order logic. Modal logic [27], many temporal logics [22], [24, Chapter 24], the guarded fragment [9], many description logics [2], and the two-variable fragment [10] are restricted first order logics with decidable finite satisfiability, and hundreds of papers on these topics have explored the border between decidability and undecidability. While many of the earlier papers exploited variations of the tree model property to show decidability, recent research has also focused on logics such as the two-variable fragment with counting C^2 [11, 23], where finite satisfiability is decidable despite the absence of the tree model property. In a recent breakthrough result, Charatonik and Witkowski [4] have extended this result to structures with built-in binary trees. Note that this logic is not a fragment of first order logic, but more naturally understood as a very weak second order logic which can express one specific second order property – the property of being a tree.

Our main result is a powerful generalization of the seminal result on decidability of the satisfiability problem of MSO over bounded tree-width and the recent theorem by [4]: We show decidability of finite satisfiability of conjunctions $\alpha \wedge \beta$ where α is in MSO and β is in C^2 by a finite structure $\mathfrak{M}$ whose restriction to the vocabulary of α has bounded tree-width. (Theorem 3.1 in Section 3)

Let us put this result into perspective:

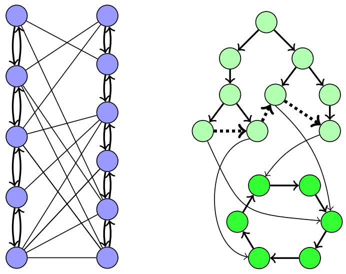
- The MSO decidability problem is a trivial consequence by setting β to true; Charatonik and Witkowski's result follows by choosing α to be an MSO formula which axiomatizes a d-ary tree, which is a standard construction [6].
- The decidability of *model checking* $\alpha \wedge \beta$ over a finite structure is a much simpler problem than ours: We just have to model check α and β one after the other. In contrast, satisfiability is not obvious because α and β can share relational variables. running two finite satisfiability algorithms for the two formulas independently may yield two models which disagree on the shared vocabulary. Thus, the problem we consider is similar in spirit to (but technically very different from) Nelson-Oppen [21] combinations of theories.
- Our result trivially generalizes to Boolean combinations of sentences in the two logics.

Proof Technique

We show how to reduce our satisfiability problem for $\alpha \wedge \beta$ to the finite satisfiability of a C^2-sentence with a built-in tree, which is decidable by [4]. The most significant technical challenge is to eliminate shared binary relation symbols between the two conjuncts. Our Separation Theorem overcomes this challenge by a construction based on local types of universe elements and a coloring argument for directed graphs. The second technical challenge is to replace the MSO-sentence α with an equi-satisfiable C^2-sentence α'. To do so, we apply tools including the Feferman-Vaught theorem for MSO and translation schemes.

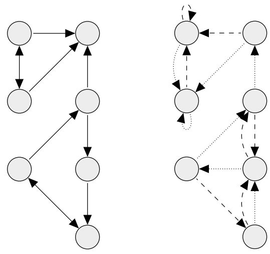
Monadic Second Order Logic with Cardinalities

Our main theorem implies new decidability results for monadic second order logic with cardinality constraints, i.e., expressions of the form $|X_1| + \ldots |X_r| < |Y_1| + \ldots |Y_t|$ where X_i and Y_i are monadic second order variables. Klaedtke and Rueß [15] showed that the decision problem for the theory of weak monadic second order logic with cardinality constraints of one



(a) (b)

Figure 1



(a) (b)

Figure 2

successor ($WS1S^{card}$) is undecidable; they described a decidable fragment where the second order quantifiers have no alternation and appear *after* the first order quantifiers in the prefix. Our main theorem implies decidability of a different fragment of WS1S with cardinalities: The fragment $MSO^{\exists card}$ consists of formulas $\exists \bar{X} \psi$ where the cardinality constraints in ψ involve only the monadic second order variables from $\bar{X}$, cf. Theorem 7.1 in Section 7. Note that in contrast to [15], our fragment is a strict superset of WS1S.

For WS2S, we are not aware of results about decidable fragments with cardinalities. We describe a strict superset of MSO whose satisfiability problem over finite graphs of bounded tree-width is decidable, and which is syntactically similar to the WS1S extension above.

Expressive Power over Structures

Our main result extends the existing body of results on finite satisfiability by structures of bounded tree-width to a significantly richer set of structures. The structures we consider are C^2-axiomatizable extensions of structures of bounded tree-width. For instance, we can have interconnected doubly-linked lists as in Fig. 1(a), or a tree whose leaves are connected in a chain and have edges pointing to any of the nodes of a cyclic list as in Fig. 1(b). Such structures occur very naturally as shapes of dynamic data structures in programming – where cycles and trees are containers for data, and additional edges express relational information between the data. The analysis of semantic relations between data structures served as a motivation for us to investigate the logics in the current paper [3].

Being a cyclic list or a tree whose leaves are chained can be expressed in MSO and both of these data structures have tree-width at most 3. We can compel the edges between the tree and the cyclic list to obey C^2-expressible constraints such as:

- every leaf of the tree has a single edge to the cyclic list;
- every node of the cyclic list has an incoming edge from at least one leaf of the tree; or
- any two leaves pointing to the same node of the cyclic list agree on membership in some unary relation.

Note that while the structures we consider may contain grids of unbounded sizes as subgraphs, the logic cannot axiomatize them.

2 Background

This section introduces basic definitions and results in model theory and graph theory. We follow [20] and [6].

The **two-variable fragment with counting** C^2 is the extension of the two-variable fragment of first order logic with first order counting quantifiers $\exists^{\leq n}$, $\exists^{\geq n}$, $\exists^{=n}$, for every $n \in \mathbb{N}$. Note that C^2 remains a fragment of first order logic. **Monadic Second order logic** MSO is the extension of first order logic with set quantifiers which can range over elements of the universe or subsets of relations[2]. Throughout the paper all structures consist of unary and binary relations only. Structures are finite unless explicitly stated otherwise (in the discussion of WS1S). Let $\mathcal{C}$ be a vocabulary (signature). The **arity** of a relation symbol $C \in \mathcal{C}$ is denoted by $arity(C)$. The set of unary (binary) relation symbols in $\mathcal{C}$ are **un($\mathcal{C}$) (bin($\mathcal{C}$))**. We write MSO($\mathcal{C}$) for the set of MSO-formulas on the vocabulary $\mathcal{C}$. The quantifier rank of a formula $\varphi \in$ MSO, i.e. the maximal depth of nested quantifiers in φ is denoted qr(φ). We denote by $\mathfrak{A}_1 \sqcup \mathfrak{A}_2$ the **disjoint union** of two $\mathcal{C}$-structures $\mathfrak{A}_1$ and $\mathfrak{A}_2$. Given vocabularies $\mathcal{C}_1 \subseteq \mathcal{C}_2$, a $\mathcal{C}_2$-structure $\mathfrak{A}_2$ is an **expansion** of a $\mathcal{C}_1$-structure $\mathfrak{A}_1$ if $\mathfrak{A}_1$ and $\mathfrak{A}_2$ agree on the symbols in $\mathcal{C}_1$; in this case $\mathfrak{A}_1$ is the **reduct** of $\mathfrak{A}_2$ to $\mathcal{C}_1$, i.e. $\mathfrak{A}_1$ is the $\mathcal{C}_1$-reduct of $\mathfrak{A}_2$. We denote the reduct of $\mathfrak{A}_2$ to $\mathcal{C}_1$ by $\mathfrak{A}_2|_{\mathcal{C}_1}$. A $\mathcal{C}$-structure $\mathfrak{A}_0$ with universe A_0 is a **substructure** of a $\mathcal{C}$-structure $\mathfrak{A}_1$ with universe A_1 if $A_0 \subseteq A_1$ and for every $R \in \mathcal{C}$, $R^{\mathfrak{A}_0} = R^{\mathfrak{A}_1} \cap A_0^{arity(R)}$. We say that $\mathfrak{A}_0$ is the substructure of $\mathfrak{A}_1$ generated by A_0.

Graphs are structures of the vocabulary[3] $\mathcal{C}_G = \langle s \rangle$ consisting of a single binary relation symbol s. Graphs are undirected without multiple edges but possibly with self-loops unless explicitly stated otherwise. **Tree-width** tw(G) is a graph parameter indicating how close a simple undirected graph G is to being a tree, cf. [6]. It is well-known that a graph has tree-width at most k iff it is a partial k-tree. A **partial k-tree** is a subgraph of a k-tree. **k-trees** are built inductively from the $(k+1)$-clique by repeated addition of vertices, each of which is connected with k edges to a k-clique. The **Gaifman graph** Gaif($\mathfrak{A}$) of a $\mathcal{C}$-structure $\mathfrak{A}$ is the graph whose vertex set is the universe of $\mathfrak{A}$ and whose edge set is the union of the symmetric closures of $C^{\mathfrak{A}}$ for every $C \in$ bin($\mathcal{C}$). Note the unary relations of $\mathfrak{A}$ play no role in Gaif($\mathfrak{A}$). The **tree-width tw($\mathfrak{A}$) of a $\mathcal{C}$-structure** $\mathfrak{A}$ is the tree-width of its Gaifman graph. In this paper, tree-width is a parameter of *finite* structures only. Fix $k \in \mathbb{N}$ for the rest of the paper. k will denote the tree-width bound we consider.

We introduce the notion of **oriented k-trees** which refines the notion of k-trees. Let $\mathcal{R} = \{R_1, \ldots, R_k\}$ be a vocabulary consisting of binary relation symbols. An oriented k-tree is an $\mathcal{R}$-structure $\mathfrak{R}$ in which all $R_i^{\mathfrak{R}}$ are total functions and whose Gaifman graph Gaif($\mathfrak{R}$) is a partial k-tree.

▶ **Lemma 2.1.** *Every $\mathcal{C}$-structure $\mathfrak{M}$ of tree-width k can be expanded into a $(\mathcal{C} \cup \mathcal{R})$-structure $\mathfrak{N}$ such that:*
- **(i)** *$\mathfrak{N}|_{\mathcal{R}}$ is an oriented k-tree,*
- **(ii)** Gaif($\mathfrak{N}$) *is a subgraph of* Gaif($\mathfrak{N}|_{\mathcal{R}}$), *and*
- **(iii)** *the tree-width of $\mathfrak{N}$ is k.*

The oriented 2-tree in Fig. 2(b) is an expansion of the directed graph in Fig. 2(a) as guaranteed in Lemma 2.1. In Fig. 2(b), R_1 and R_2 are denoted by the dashed arrows and the dotted arrows, respectively. There are several other oriented k-trees which expand Fig. 2(a) and fulfill the requirements in Lemma 2.1,

To see that Lemma 2.1 holds, we describe a construction of $\mathfrak{N}$ echoing the process of constructing k-trees above. For each vertex u of the initial $(k+1)$-clique, we can set the

[2] On relational structures, MSO is also known as *Guarded Second Order logic* GSO. The results of this paper extend to CMSO, the extension of MSO with modular counting quantifiers.

[3] Since we explicitly allowed quantification over subsets of relations for MSO, we do not view graphs and structures as incidence structures, in contrast to [6, Sections 1.8.1 and 1.9.1].

values of $R_1^{\mathfrak{N}}(u), \ldots, R_k^{\mathfrak{N}}(u)$ to be the other k vertices of the clique. When a new vertex u is added to the k-tree, k edges incident to it are added. We set $R_1^{\mathfrak{N}}(u), \ldots, R_k^{\mathfrak{N}}(u)$ to be the set of vertices incident to u. For oriented k-trees whose Gaifman graph is not a k-tree the construction of an oriented k-tree is augmented by changing the value of $R_i^{\mathfrak{N}}(u)$ to $R_i^{\mathfrak{N}}(u) = u$ whenever $R_i^{\mathfrak{N}}(u)$ is not well-defined. This can happen when the target of u under $R_i^{\mathfrak{N}}$ is a vertex which was eliminated by taking the subgraph of a k-tree to obtain the partial k-tree.

3 Overview of the Main Theorem and its Proof

The precise statement of the main theorem is as follows:

▶ **Theorem 3.1** (Main Theorem). *Let $\mathcal{C}_{bnd}$ and $\mathcal{C}_{unb}$ be vocabularies. Let s be a binary relation symbol not in $\mathcal{C}_{bnd} \cup \mathcal{C}_{unb}$. Let $\alpha \in \mathrm{MSO}(\mathcal{C}_{bnd})$ and $\beta \in C^2(\mathcal{C}_{unb})$. There is an effectively computable sentence $\delta \in C^2(\mathcal{D})$ over a vocabulary $\mathcal{D} \supseteq \{s\}$ such that the following are equivalent:*

(i) *There is a $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$-structure $\mathfrak{M}$ such that $\mathfrak{M} \models \alpha \wedge \beta$ and $\mathrm{tw}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) \leq k$.*

(ii) *There is a $\mathcal{D}$-structure $\mathfrak{N}$ such that $\mathfrak{N} \models \delta$ and $s^{\mathfrak{N}}$ is a binary tree.*

The first step towards proving Theorem 3.1 is the Separation Theorem:

▶ **Theorem 3.2** (Separation Theorem). *Let $\mathcal{C}_{bnd}$ and $\mathcal{C}_{unb}$ be vocabularies. Let $\alpha \in \mathrm{MSO}(\mathcal{C}_{bnd})$ and $\beta \in C^2(\mathcal{C}_{unb})$. There are effectively computable sentences $\alpha' \in \mathrm{MSO}(\mathcal{D}_{bnd})$ and $\beta' \in C^2(\mathcal{D}_{unb})$ over vocabularies $\mathcal{D}_{bnd}$ and $\mathcal{D}_{unb}$ such that $\mathcal{D}_{bnd} \cap \mathcal{D}_{unb}$ only contains unary relation symbols and the following are equivalent:*

(i) *There is a $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$-structure $\mathfrak{M}$ with $\mathfrak{M} \models \alpha \wedge \beta$ and $\mathrm{tw}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) \leq k$*

(ii) *There is a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure $\mathfrak{N}$ with $\mathfrak{N} \models \alpha' \wedge \beta'$ and $\mathrm{tw}(\mathfrak{N}|_{\mathcal{D}_{bnd}}) \leq k$.*

In conjunction with Theorem 3.2, we only need to prove Theorem 3.1 in the case that the MSO-formula α and the C^2-formula β only share unary relation symbols. The significance of Theorem 3.2 is that it allows us to use tools designed for MSO in our more involved setting. The proof of Theorem 3.2 uses notions of types for C^2-sentences in Scott normal form, coloring arguments, and an induction on ranks of structures. Theorem 3.2 is discussed in Section 4. The next step is to move from structures whose reducts have bounded tree-width to structures which contain a binary tree.

▶ **Lemma 3.3.** *Let $\mathcal{C}_{bnd}$ and $\mathcal{C}_{unb}$ be vocabularies such that $\mathcal{C}_{bnd} \cap \mathcal{C}_{unb}$ contains only unary relation symbols. Let s be a binary relation symbol. There is a vocabulary $\mathcal{D}_{bnd}$ consisting of s and unary relation symbols only as follows. For every $\alpha \in \mathrm{MSO}(\mathcal{C}_{bnd})$ and $\beta \in C^2(\mathcal{C}_{unb})$, there are effectively computable sentences $\alpha' \in \mathrm{MSO}(\mathcal{D}_{bnd})$ and $\beta' \in C^2(\mathcal{D}_{bnd} \cup \mathcal{C}_{unb})$ such that the following are equivalent:*

(i) *There is a $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$-structure $\mathfrak{M}$ such that $\mathfrak{M} \models \alpha \wedge \beta$ and $\mathrm{tw}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) \leq k$.*

(ii) *There is a $(\mathcal{D}_{bnd} \cup \mathcal{C}_{unb})$-structure $\mathfrak{N}$ such that $\mathfrak{N} \models \alpha' \wedge \beta'$ and $s^{\mathfrak{N}}$ is a binary tree.*

Technically, Lemma 3.3 is proved using a translation scheme which maps structures with a binary tree into structures whose $\mathcal{C}_{bnd}$-reducts have tree-width at most k, and conversely, each of the latter structures is the image of a structure with a binary tree under the translation scheme. Translation schemes capturing the graphs of tree-width at most k as the image of labeled trees were studied in the context of decidability and model checking of MSO [1]. We need a more refined construction to ensure that the translation scheme also behaves correctly on C^2-sentences, i.e. that it maps C^2-sentences to C^2-sentences, see Lemma 3.3 in Section 5.

Now that we have reduced our attention to the case that our structures contain a binary tree, we can replace MSO-sentences with equi-satisfiable C^2-sentences.

▶ **Lemma 3.4.** *Let $\mathcal{C}$ be a vocabulary which consists only of a binary relation symbol s and unary relation symbols. Let α be an* MSO($\mathcal{C}$)*-sentence. There is an effectively computable $C^2(\mathcal{D})$-sentence γ over a vocabulary $\mathcal{D} \supseteq \mathcal{C}$ such that for every $\mathcal{C}$-structure $\mathfrak{M}$ in which $s^{\mathfrak{M}}$ is a binary tree the following are equivalent:*
(i) $\mathfrak{M} \models \alpha$.
(ii) *There is a $\mathcal{D}$-structure $\mathfrak{N}$ expanding $\mathfrak{M}$ such that $\mathfrak{N} \models \gamma$.*

For the proof of Lemma 3.4 we use a Feferman-Vaught type theorem which states that the Hintikka type (i.e. MSO types) of a binary tree labeled with unary relation symbols depends only on the Hintikka types of its children. We can therefore axiomatize in C^2 that the Hintikka type of the labeled binary tree implies a given MSO-sentence.

Having replaced the MSO-sentence in statement (ii) of Lemma 3.3 with a C^2-sentence, we are left with the problem of deciding whether a C^2-sentence is satisfiable by a structure in which a specified relation is a binary tree, which has recently been shown to be decidable:

▶ **Theorem 3.5** (Charatonik and Witkowski [4])**.** *Let $\mathcal{C}$ be a vocabulary which contains a binary relation symbol s. Given a $C^2(\mathcal{C})$-sentence φ, it is decidable whether φ is satisfiable by a structure $\mathfrak{M}$ in which $s^{\mathfrak{M}}$ is a binary tree.*

4 Separation Theorem

4.1 Basic Definitions and Results

1-types and 2-types

We begin with some notation and definitions in the spirit of the literature on decidability of C^2, cf. e.g. [23, 4]. Let $\mathcal{A}$ be a vocabulary of unary and binary relations.

A **1-type** π is a maximal consistent set of atomic $\mathcal{A}$-formulas or negations of atomic $\mathcal{A}$-formulas with free variable x, i.e., exactly one of $A(x)$ and $\neg A(x)$ belongs to π for every unary relation symbol $A \in \mathcal{A}$, and exactly one of $B(x,x)$ and $\neg B(x,x)$ belongs to π for every binary relation symbol $B \in \mathcal{A}$. We denote by $\mathfrak{char}_\pi(x) = \bigwedge_{\iota\in\pi} \iota$ the formula that characterizes the 1-type π. We denote by 1-Types($\mathcal{A}$) the set of 1-types over $\mathcal{A}$.

A **2-type** λ is a maximal consistent set of atomic $\mathcal{A}$-formulas or negations of atomic $\mathcal{A}$-formulas with free variables x and y and $x \not\approx y \in \lambda$, i.e., for every $z \in \{x, y\}$ and unary relation symbol $A \in \mathcal{A}$, exactly one of $A(z)$ and $\neg A(z)$ belongs to λ, and for every $z_1, z_2 \in \{x, y\}$ and binary relation symbol $B \in \mathcal{A}$, exactly one of $B(z_1, z_2)$ and $\neg B(z_1, z_2)$ belongs to λ. We write λ^{-1} for the 2-type obtained from λ by substituting all occurrences of x resp. y with y resp. x. We write λ_x for the 1-type obtained from λ by restricting λ to formulas with free variable x. We write λ_y for the 1-type obtained from λ by restricting λ to formulas with free variable y *and substituting y with x*. We denote by $\mathfrak{char}_\lambda(x,y) = \bigwedge_{\iota\in\lambda} \iota$ the formula that characterizes the 2-type λ. We denote by 2-Types($\mathcal{A}$) the set of 2-types over $\mathcal{A}$.

Let $\mathfrak{M}$ be an $\mathcal{A}$-structure. We denote by 1-$\mathfrak{tp}^{\mathfrak{M}}(u)$ the unique 1-type π such that $\mathfrak{M} \models \mathfrak{char}_\pi(u)$. For elements u, v of $\mathfrak{M}$, we denote by 2-$\mathfrak{tp}^{\mathfrak{M}}(u,v)$ the unique 2-type λ such that $\mathfrak{M} \models \mathfrak{char}_\lambda(u,v)$. We denote by 2-$\mathfrak{tp}(\mathfrak{M}) = \{$2-$\mathfrak{tp}^{\mathfrak{M}}(u,v) \mid u, v$ elements of $\mathfrak{M}\}$ the set of 2-types *realized* by $\mathfrak{M}$. The following lemma is easy to see:

▶ **Lemma 4.1.** *Let $\mathfrak{M}_1, \mathfrak{M}_2$ be two $\mathcal{A}$-structures over the same universe M and let $\phi = \forall x, y.\, \chi \in C^2(\mathcal{A})$ with χ quantifier-free. If* 2-$\mathfrak{tp}(\mathfrak{M}_1) =$ 2-$\mathfrak{tp}(\mathfrak{M}_2)$*, then $\mathfrak{M}_1 \models \phi$ iff $\mathfrak{M}_2 \models \phi$.*

Scott Normal Form and $\mathcal{T}$-functionality

C^2-sentences have a Scott-Normal Form, cf. [12], which can be obtained by iteratively applying Skolemization and introducing new predicates for subformulas, together with predicates ensuring the soundness of this transformation:

▶ **Lemma 4.2** (Scott Normal Form, [12]). *For every C^2-sentence β there is a C^2-sentence β' of the form*

$$\forall x, y.\, \chi \wedge \bigwedge_{i\in[l]} \forall x.\, \exists^{=1} y.\, S_i(x, y), \tag{1}$$

with χ quantifier-free, over an expanded vocabulary such that β and β' are equi-satisfiable. Moreover, β' is computable. The expanded vocabulary contains in particular the fresh binary relation symbols $\mathcal{S} = \{S_1, \ldots, S_l\}$.

Let $\mathcal{T}$ be a set of binary relation symbols. We say a structure $\mathfrak{M}$ is **$\mathcal{T}$-functional**, if for every $T \in \mathcal{T}$, $T^{\mathfrak{M}}$ is a total function on the universe of $\mathfrak{M}$. Observe the following are equivalent for every structure $\mathfrak{M}$:

(i) $\mathfrak{M}$ satisfies Eq. (1), and

(ii) $\mathfrak{M} \models \forall x, y.\, \chi$ and $\mathfrak{M}$ is $\mathcal{S}$-functional.

Message Types and Chromaticity

Let $\mathcal{T} \subseteq \mathrm{bin}(\mathcal{A})$ be a subset of the binary relation symbols of $\mathcal{A}$. We write $\lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{A})$ and say λ is a **$\mathcal{T}$-message type**, if $\lambda \in 2\text{-Types}(\mathcal{A})$ and $T(x, y) \in \lambda$ for some $T \in \mathcal{T}$. Let $\mathfrak{M}$ be a $\mathcal{A}$-structure with universe M. We define $E = \{(u, v) \in M^2 \mid \text{there is a } T \in \mathcal{T} \text{ with } \mathfrak{M} \models T(u, v)\}$. The **$\mathcal{T}$-message-graph** is the directed graph $G = (M, F)$, where $F = \{(u, v) \in M^2 \mid u \neq v \text{ and } (u, v) \in E \circ E\}$, where $R \circ S = \{(a, b) \mid \text{ there is a } c \text{ with } (a, c) \in R \text{ and } (c, b) \in S\}$ denotes the usual *composition* of relations. We say $\mathfrak{M}$ is **$\mathcal{T}$-chromatic**, if $\text{1-tp}^{\mathfrak{M}}(u) \neq \text{1-tp}^{\mathfrak{M}}(v)$ for all $(u, v) \in F$.

We note that if $\mathfrak{M}$ is $\mathcal{T}$-functional, then G has out-degree $deg^+(u) \leq |\mathcal{T}|^2$ for all $u \in M$. This allows us to prove Lemma 4.3 based on Lemma 4.4.

▶ **Lemma 4.3.** *There is a finite set of unary relations symbols $\mathrm{colors}(\mathcal{T})$ such that every $\mathcal{T}$-functional $\mathcal{A}$-structure can be expanded to a $\mathcal{T}$-chromatic $(\mathcal{A} \cup \mathrm{colors}(\mathcal{T}))$-structure.*

▶ **Lemma 4.4.** *Let $G = (V, E)$ be a directed graph with out-degree $deg^+(v) \leq k$ for all $v \in V$. Then, the underlying undirected graph has a proper $(2k + 1)$-coloring.*

4.2 Separation Theorem

Let $G = (V, E)$ be an (undirected) graph. We say G is **k-bounded**, if the edges of G can be oriented such that every node of G has out-degree less than k. We say a structure $\mathfrak{M}$ is k-bounded if its Gaifman graph is k-bounded. We note that the formulation of separation theorem below is slightly more general than the formulation in Section 3 because it is stated in terms of k-bounded graphs (graphs with tree-width k are clearly k-bounded, see the discussion on oriented k-trees in Section 2).

▶ **Theorem 3.2** (Separation Theorem). *Let k be a natural number. Let $\mathcal{C}_{bnd}$ and $\mathcal{C}_{unb}$ be vocabularies. Let $\alpha \in \mathrm{MSO}(\mathcal{C}_{bnd})$ and[4] $\beta \in C^2(\mathcal{C}_{unb})$. There are effectively computable*

[4] The Separation Theorem remains correct if we replace C^2 with any logic containing C^2 which is closed under conjunction.

sentences $\alpha' \in \mathrm{MSO}(\mathcal{D}_{bnd})$ *and* $\beta' \in C^2(\mathcal{D}_{unb})$ *over vocabularies* $\mathcal{D}_{bnd}$ *and* $\mathcal{D}_{unb}$ *such that* $\mathcal{D}_{bnd} \cap \mathcal{D}_{unb}$ *only contains unary relation symbols such that for every* k*-bounded graph* G *the following are equivalent:*

(i) *There is a* $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$*-structure* $\mathfrak{M}$ *with* $\mathfrak{M} \models \alpha \wedge \beta$ *and* $\mathrm{Gaif}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) = G$.

(ii) *There is a* $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$*-structure* $\mathfrak{N}$ *with* $\mathfrak{N} \models \alpha' \wedge \beta'$ *and* $\mathrm{Gaif}(\mathfrak{N}|_{\mathcal{D}_{bnd}}) = G$.

We assume that β is in the form given in Eq. (1) for some set of binary relation symbols $\mathcal{S} = \{S_1, \ldots, S_l\} \subseteq \mathcal{C}_{unb}$ and quantifier-free C^2-formula χ. Let $\mathcal{R} = \{R_1, \ldots, R_k\}$ be a set of fresh binary relation symbols. We set $\mathcal{T} = \mathcal{S} \cup \mathcal{R}$. We begin by giving an intuition for the proof of the Separation Theorem in three stages.

4.2.1 Syntactic separation coupled with semantic constraints

For a binary relation symbol B, we define its *copy* as the relation symbol $\overline{B}$. For every vocabulary $\mathcal{A}$, we define its *copy* $\overline{\mathcal{A}} = \mathrm{un}(\mathcal{A}) \cup \{\overline{B} \mid B \in \mathcal{A}\}$ to be the unary relation symbols of $\mathcal{A}$ plus the copies of its binary relations symbols. We assume that copied relation symbols are distinct from non-copied symbols, i.e., $\mathrm{bin}(\mathcal{A}) \cap \mathrm{bin}(\overline{\mathcal{A}}) = \emptyset$. For a formula φ over vocabulary $\mathcal{A}$, we define its *copy* $\overline{\varphi}$ over vocabulary $\overline{\mathcal{A}}$ as the formula obtained from φ by substituting every occurrence of a binary relation symbol $B \in \mathcal{A}$ with $\overline{B}$.

The sentences $\overline{\alpha}$ (the copy of α) and β do not share any binary relation symbols. Clearly, (i) from Theorem 3.2 holds iff

(I) $\overline{\alpha} \wedge \beta$ is satisfied by a $((\overline{\mathcal{C}_{bnd}} \cup \overline{\mathcal{C}_{unb}}) \cup (\mathcal{C}_{bnd} \cup \mathcal{C}_{unb}))$-structure $\mathfrak{N}$ with $B^{\mathfrak{N}} = \overline{B}^{\mathfrak{N}}$ for all $B \in \mathrm{bin}(\mathcal{C}_{bnd})$ and $\mathrm{Gaif}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) = G$.

In the next two stages we will construct α' and β' so that (I) is equivalent to (ii) from Theorem 3.2. More precisely, we will construct sentences $\mu_{bnd}, \mu_{unb} \in C^2(\mathcal{D}_{unb})$ with $\mathcal{D}_{unb} \supseteq \mathcal{C}_{bnd} \cup \mathcal{C}_{unb}$ and $\mathcal{D}_{bnd} = \overline{\mathcal{D}_{unb}}$ such that (I) is equivalent (II):

(II) $(\overline{\alpha} \wedge \overline{\mu_{bnd}}) \wedge (\beta \wedge \mu_{unb})$ is satisfied by a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure $\mathfrak{N}$ with $\mathrm{Gaif}(\mathfrak{N}|_{\mathcal{D}_{bnd}}) = G$.

4.2.2 Representation of k-bounded structures using functions and unary relations

k-bounded $\mathcal{A}$-structures $\mathfrak{A}$ can be represented by introducing new binary relation symbols interpreted as functions and new unary relation symbols as follows.

(a) We add k fresh relation symbols $\mathcal{R} = \{R_1, \ldots, R_k\}$ and axiomatize that these relations are interpreted as total functions.

(b) We add fresh unary relations P_λ for each $\mathcal{R}$-message type and axiomatize that every element labeled by P_λ has an outgoing edge with 2-type λ. The symbols P_λ are called *unary* 2*-type annotations.*

(c) We axiomatize that $\mathrm{Gaif}(\mathfrak{A}|_{\mathrm{bin}(\mathcal{A})}) = \mathrm{Gaif}(\mathfrak{A}|_{\mathcal{R}})$.

In other words, the functions interpreting $R_1, \ldots, R_k$ witness that $\mathfrak{A}$ can be oriented so that every node in the Gaifman graph of $\mathfrak{A}$ has outdegree at most k. The 2-type of each edge (u, v) in $\mathfrak{A}$ is encoded by putting the unary relation symbol P_λ of the 2-type of (u, v) on the source u in the orientation.

Theorem 3.2 as well as (I) and (II) involve reducts which are k-bounded structures. Given a $((\overline{\mathcal{C}_{bnd}} \cup \overline{\mathcal{C}_{unb}}) \cup (\mathcal{C}_{bnd} \cup \mathcal{C}_{unb}))$-structure $\mathfrak{N}$, we will use the above representation twice, on $\mathfrak{N}|_{\mathcal{C}_{bnd}}$ and $\mathfrak{N}|_{\overline{\mathcal{C}_{bnd}}}$, by axiomatizing that every element labeled by P_λ has an outgoing edge with 2-type λ and an outgoing edges with 2-type $\overline{\lambda}$. This allows us to replace the condition from (I) that $B^{\mathfrak{N}} = \overline{B}^{\mathfrak{N}}$ for all $B \in \mathrm{bin}(\mathcal{C}_{bnd})$ with the condition that $R_i^{\mathfrak{N}} = \overline{R_i}^{\mathfrak{N}}$ for all $R_i \in \mathcal{R}$.

4.2.3 Establishing the semantic condition of (I) by swapping edges

Here we discuss how to show the implication from (II) to (I). Let $\mathfrak{N}$ be a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure with $\mathfrak{N} \models (\overline{\alpha} \wedge \overline{\mu_{bnd}}) \wedge (\beta \wedge \mu_{unb})$. It simplifies the discussion to split a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure $\mathfrak{N}$ into two $\mathcal{D}_{unb}$-structures. The $\mathcal{D}_{unb}$-structure $\mathfrak{L}$ is $\mathfrak{N}|_{\mathcal{D}_{unb}}$. The $\mathcal{D}_{unb}$-structure $\mathfrak{L}'$ is obtained from $\mathfrak{N}|_{\mathcal{D}_{bnd}}$ by renaming copies of relation symbols $\overline{B}$ to B — i.e., we define the $\mathcal{D}_{unb}$-structure $\mathfrak{L}'$ by setting $1\text{-}\mathfrak{tp}^{\mathfrak{L}'}(u) = 1\text{-}\mathfrak{tp}^{\mathfrak{N}}(u)$ for all $u \in M$ and setting $\mathfrak{L}' \models B(u, v)$ iff $\mathfrak{N} \models \overline{B}(u, v)$ for all $u, v \in M$ and $B \in \mathrm{bin}(\mathcal{D}_{bnd})$. The interpretations of the relations R_i might differ in $\mathfrak{L}$ and $\mathfrak{L}'$. Observe that we have $\mathfrak{L}' \models \alpha$ and $\mathfrak{L} \models \beta$. The *key idea of the proof* is to show the existence of a sequence of structures $\mathfrak{L} = \mathfrak{L}_0, \ldots, \mathfrak{L}_p$, where each $\mathfrak{N}_{i+1}$ is obtained from $\mathfrak{N}_i$ by *swapping edges*, until the interpretations of the relations R_i agree in $\mathfrak{L}_p$ and $\mathfrak{L}'$. The edge swapping operation is a local operation which involves changing the 2-types of at most 4 edges.

The edge swapping operation satisfies two crucial preservation requirements: edge swapping preserves (PR-1) the truth value of β, i.e. $\mathfrak{L}_p \models \beta$, and (PR-2) $\mathcal{R}$-functionality. The universal constraint $\forall x, y.\, \chi$ in β is maintained under edge swapping because this operation does not change the set of 2-types (see Lemma 4.1). To satisfy the preservation requirements (PR-1) and (PR-2), all that remains is to guarantee the existence of a sequence of edge swapping preserving $\mathcal{S}$-functionality and $\mathcal{R}$-functionality (which amounts to $\mathcal{T}$-functionality because of $\mathcal{T} = \mathcal{S} \cup \mathcal{R}$). We use two main techniques that guarantee the existence of edges for which edge swapping preserves $\mathcal{T}$-functionality: chromaticity and unary 2-type annotations. We will axiomatize that the structures $\mathfrak{L}$ and $\mathfrak{L}'$ are chromatic; we will take care that chromaticity is maintained during edge swaps. We will add fresh unary relation symbols P_λ for every $\mathcal{T}$-message type λ and axiomatize that every element of $\mathfrak{N}$ labeled by P_λ has an outgoing edge with 2-type λ and an outgoing edge with 2-type $\overline{\lambda}$; we will take care that such outgoing edges are maintained during edge swaps.

4.2.4 Proof of the Separation Theorem

We now start the formal proof of the Separation Theorem. Let colors($\mathcal{T}$) be the vocabulary from Lemma 4.3. We set $\mathcal{E} = \mathcal{C}_{bnd} \cup \mathcal{C}_{unb} \cup \mathcal{R} \cup \mathrm{colors}(\mathcal{T})$ and $\mathcal{P} = \{P_\lambda \mid \lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{E})\}$. We set $\mathcal{D}_{unb} = \mathcal{E} \cup \mathcal{P}$ and $\mathcal{D}_{bnd} = \overline{\mathcal{D}_{unb}}$. Next we will define formulas $\alpha^\circ \in \mathrm{MSO}(\mathcal{D}_{unb})$ and $\beta' \in C^2(\mathcal{D}_{unb})$, and set $\alpha' = \overline{\alpha^\circ} \in \mathrm{MSO}(\mathcal{D}_{bnd})$. We set
$\mu_{bnd} = \psi_{functionality} \wedge \psi_{gaifmann} \wedge \psi_{typesBnd} \wedge \psi_{chromaticity} \wedge \psi_{subgraph}$, $\alpha^\circ = \alpha \wedge \mu_{bnd}$,
$\mu_{unb} = \psi_{functionality} \wedge \psi_{gaifmann} \wedge \psi_{typesUnb} \wedge \psi_{chromaticity}$ and $\beta' = \beta \wedge \mu_{unb}$, where:

- $\psi_{functionality} = \bigwedge_{i \in [k]} \forall x.\, \exists^{=1} y.\, R_i(x, y)$
  The formula $\psi_{functionality}$ expresses that each R_i is interpreted as a total function.
- $\psi_{gaifmann} = \forall x, y.\, x \not\approx y \to \left(\bigvee_{i \in [k]} R_i(x, y) \vee R_i(y, x) \leftrightarrow \bigvee_{B \in \mathrm{bin}(\mathcal{C}_{bnd})} B(x, y) \vee B(y, x)\right)$
  The formula $\psi_{gaifmann}$ expresses that for every $\mathcal{D}_{unb}$-structure $\mathfrak{L}$ with $\mathfrak{L} \models \psi_{gaifmann}$ we have that $\mathrm{Gaif}(\mathfrak{L}|_{\mathcal{R}}) = \mathrm{Gaif}(\mathfrak{L}|_{\mathcal{C}_{bnd}})$.
- $\psi_{subgraph} = \bigwedge_{B \in \mathrm{bin}(\mathcal{C}_{unb})} \forall x, y.\, x \not\approx y \to \left(B(x, y) \to \bigvee_{i \in [k]} R_i(x, y) \vee R_i(y, x)\right)$
  The formula $\psi_{subgraph}$ expresses that for every $\mathcal{D}_{unb}$-structure $\mathfrak{L}$ with $\mathfrak{L} \models \psi_{subgraph}$ we have that $\mathrm{Gaif}(\mathfrak{L}|_{\mathcal{C}_{unb}})$ is a subgraph of $\mathrm{Gaif}(\mathfrak{L}|_{\mathcal{R}})$.
- $$\psi_{typesUnb} = \bigwedge_{\lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{E})} \forall x.\, P_\lambda(x) \leftrightarrow \exists y. \mathfrak{char}_\lambda(x, y)$$

  The formula $\psi_{typesUnb}$ expresses that for every $\mathcal{T}$-message type λ a node u satisfies the predicate P_λ iff u has an outgoing edge with 2-type λ.

- $\psi_{typesBnd} = \bigwedge_{\substack{\lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{E}), i \in [k]: \\ R_i(x,y) \in \lambda \text{ or } R_i(y,x) \in \lambda}} \forall x. P_\lambda(x) \leftrightarrow \exists y.\mathfrak{char}_\lambda(x,y)$

  The formula $\psi_{typesBnd}$ expresses that for every $\mathcal{T}$-message type λ, where λ contains the predicate $R_i(x,y)$ or $R_i(y,x)$ for some $i \in [k]$, a node u satisfies the predicate P_λ iff u has an outgoing edge with 2-type λ.
- $\psi_{chromaticity} = \bigwedge_{\substack{\lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{E}), T \in \mathcal{T}: \\ \neg T(y,x) \in \lambda}} \forall x, y. P_\lambda(x) \wedge T(y,x) \rightarrow \neg\mathfrak{char}_{\lambda_y}(y)$

  The formula $\psi_{chromaticity}$ expresses that for every $\mathcal{D}_{unb}$-structure $\mathfrak{L}$ with $\mathfrak{L} \models \psi_{typesUnb}$, $\mathfrak{L}$ is chromatic iff $\mathfrak{L} \models \psi_{chromaticity}$.

The direction "(i) implies (ii)" of the Separation Theorem is straightforward to show by appropriately expanding the model of (i) to a model of (ii):

▶ **Lemma 4.5.** *Let G be a k-bounded graph. Let $\mathfrak{M}$ be a $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$-structure with $\mathfrak{M} \models \alpha \wedge \beta$ and $\mathrm{Gaif}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) = G$. $\mathfrak{M}$ can be expanded to a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure $\mathfrak{N}$ with $\mathfrak{N} \models \alpha' \wedge \beta'$ and $\mathrm{Gaif}(\mathfrak{N}|_{\mathcal{D}_{bnd}}) = G$.*

Proof. Because of $\mathrm{Gaif}(\mathfrak{M}|_{\mathcal{C}_{bnd}}) = G$ and G is k-bounded, we can expand $\mathfrak{M}$ to a $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb} \cup \mathcal{R})$-structure $\widehat{\mathfrak{M}}$ such that $\mathrm{Gaif}(\mathfrak{L}|_{\mathcal{R}}) = \mathrm{Gaif}(\mathfrak{L}|_{\mathcal{C}_{bnd}})$ and $R_i^{\widehat{\mathfrak{M}}}$ is a total function for all $i \in [k]$ (possibly adding self-loops for the relations R_i). Thus, $\widehat{\mathfrak{M}} \models \psi_{functionality} \wedge \psi_{gaifmann}$. According to Lemma 4.3, $\widehat{\mathfrak{M}}$ can be expanded to a chromatic structure $\widetilde{\mathfrak{M}}$ over vocabulary $\mathcal{E}$ with $\widetilde{\mathfrak{M}} \models \psi_{chromaticity}$. We expand $\widetilde{\mathfrak{M}}$ to a $\mathcal{D}_{unb}$-structure $\mathfrak{L}$ such that for all $u \in M$ and $\lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{E})$ we have $\mathfrak{L} \models P_\lambda(u)$ iff there is an element v of $\mathfrak{L}$ such that $2\text{-}\mathsf{tp}^{\mathfrak{L}|_\mathcal{E}}(u,v) = \lambda$. This definition gives us $\mathfrak{L} \models \psi_{typesUnb}$, and thus $\mathfrak{L} \models \beta'$. We expand $\mathfrak{L}$ to a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure $\mathfrak{N}$ such that for all $u, v \in M$ and $B \in \mathrm{bin}(\mathcal{D}_{unb})$ we have $\mathfrak{N} \models \overline{B}(u,v)$ iff $\mathfrak{N} \models B(u,v)$ and $\mathfrak{N} \models R_i(u,v)$ or $\mathfrak{N} \models R_i(v,u)$ for some $i \in [k]$. We note that $\mathfrak{N} \models \alpha'$. ◀

Now we turn to the direction "(ii) implies (i)". Let G be a k-bounded graph. Let $\mathfrak{N}$ be a $(\mathcal{D}_{bnd} \cup \mathcal{D}_{unb})$-structure with $\mathfrak{N} \models \alpha' \wedge \beta'$ and $\mathrm{Gaif}(\mathfrak{N}|_{\mathcal{D}_{bnd}}) = G$. Let M be the universe of $\mathfrak{N}$. We define the $\mathcal{D}_{unb}$-structure $\mathfrak{L}'$ by setting $1\text{-}\mathsf{tp}^{\mathfrak{L}'}(u) = 1\text{-}\mathsf{tp}^{\mathfrak{N}}(u)$ for all $u \in M$ and setting $\mathfrak{L}' \models B(u,v)$ iff $\mathfrak{N} \models \overline{B}(u,v)$ for all $u, v \in M$ and $B \in \mathrm{bin}(\mathcal{D}_{bnd})$. We note that $\mathfrak{L}' \models \alpha^\circ$ and $\mathrm{Gaif}(\mathfrak{L}') = G$. We define the $\mathcal{D}_{unb}$-structure $\mathfrak{L}$ by setting $\mathfrak{L} = \mathfrak{N}|_{\mathcal{D}_{unb}}$. We note that $\mathfrak{L} \models \beta'$.

We make the following definition: For $u \in M$ and $i \in [k]$ we set $rank_u^i(\mathfrak{L}, \mathfrak{L}') = 1$, if there are $v, w \in M$ with $\mathfrak{L} \models R_i(u,v)$, $\mathfrak{L}' \models R_i(u,w)$ and $v \neq w$; we set $rank_u^i(\mathfrak{L}, \mathfrak{L}') = 0$, otherwise. We set $rank_u(\mathfrak{L}, \mathfrak{L}') = \sum_{i \in [k]} rank_u^i(\mathfrak{L}, \mathfrak{L}')$ and $rank(\mathfrak{L}, \mathfrak{L}') = \sum_{u \in M} rank_u(\mathfrak{L}, \mathfrak{L}')$. *rank* measures the deviation of the relations $\mathcal{R}$ in $\mathfrak{L}$ and $\mathfrak{L}'$ (we note that there always are unique $v, w \in M$ for $u \in M$ with $\mathfrak{L} \models R_i(u,v)$, $\mathfrak{L}' \models R_i(u,w)$ because of $\mathfrak{L} \models \psi_{functionality}$ and $\mathfrak{L}' \models \psi_{functionality}$). *rank* has the following important property that relates the relations R_i with the 2-types of $\mathfrak{L}$ and $\mathfrak{L}'$:

▶ **Lemma 4.6.** *Let $u \in M$ be an element with $rank_u^i(\mathfrak{L}, \mathfrak{L}') = 0$ and let $\lambda \in$ 2-Types$(\mathcal{E})$ with $R_i \in \lambda$ for some $i \in [k]$. For all $v \in M$ we have $2\text{-}\mathsf{tp}^{\mathfrak{L}|_\mathcal{E}}(u,v) = \lambda$ iff $2\text{-}\mathsf{tp}^{\mathfrak{L}'|_\mathcal{E}}(u,v) = \lambda$.*

Proof. Let $v \in M$ be an element with $2\text{-}\mathsf{tp}^{\mathfrak{L}|_\mathcal{E}}(u,v) = \lambda$. We get $\mathfrak{L} \models P_\lambda(u)$ because of $\mathfrak{L} \models \psi_{typesUnb}$. Because of $1\text{-}\mathsf{tp}^{\mathfrak{L}}(u) = 1\text{-}\mathsf{tp}^{\mathfrak{L}'}(u)$ we have $\mathfrak{L}' \models P_\lambda(u)$. Because of $\mathfrak{L}' \models \psi_{typesBnd}$ we have $\lambda = 2\text{-}\mathsf{tp}^{\mathfrak{L}'|_\mathcal{E}}(u,w)$ for some $w \in M$. $rank_u^i(\mathfrak{L}, \mathfrak{L}') = 0$ implies that $\mathfrak{L}' \models R_i(u,v)$. Because of $\mathfrak{L}' \models \psi_{functionality}$ we get $v = w$. In the same way one can show that $2\text{-}\mathsf{tp}^{\mathfrak{L}'|_\mathcal{E}}(u,v) = \lambda$ implies $2\text{-}\mathsf{tp}^{\mathfrak{L}|_\mathcal{E}}(u,v) = \lambda$. ◀

If there is no deviation of the relations $\mathcal{R}$ in $\mathfrak{L}$ and $\mathfrak{L}'$, i.e., if $rank(\mathfrak{L},\mathfrak{L}') = 0$, then we have established the direction "(ii) implies (i)":

▶ **Lemma 4.7.** *If $rank(\mathfrak{L},\mathfrak{L}') = 0$, then $\mathfrak{L}|_{\mathcal{C}_{bnd}\cup\mathcal{C}_{unb}} \models \alpha \wedge \beta$ and $\mathrm{Gaif}(\mathfrak{L}_{\mathcal{C}_{bnd}}) = G$.*

Proof. We show $\mathfrak{L}|_{\mathcal{C}_{bnd}} = \mathfrak{L}'|_{\mathcal{C}_{bnd}}$. Let $u,v \in M$ with $\mathfrak{L} \models B(u,v)$ for some $B \in \mathrm{bin}(\mathcal{C}_{bnd})$. If $u = v$ we get $\mathfrak{L}' \models B(u,v)$ because u has the same 1-type in $\mathfrak{L}$ and $\mathfrak{L}'$. We assume $u \neq v$ in the following. Because of $\mathfrak{L} \models \psi_{gaifmann}$ we have $\mathfrak{L} \models R_i(u,v)$ or $\mathfrak{L} \models R_i(v,u)$ for some $i \in [k]$. Because of $rank(\mathfrak{L},\mathfrak{L}') = 0$ we have $rank^i_u(\mathfrak{L},\mathfrak{L}') = 0$. By Lemma 4.6 we have $2\text{-}\mathfrak{tp}^{\mathfrak{L}|\mathcal{E}}(u,v) = 2\text{-}\mathfrak{tp}^{\mathfrak{L}'|\mathcal{E}}(u,v)$. Thus, $\mathfrak{L}' \models B(u,v)$. In the same way one can show that $\mathfrak{L}' \models B(u,v)$ implies $\mathfrak{L} \models B(u,v)$. Thus, $\mathfrak{L}|_{\mathcal{C}_{bnd}} = \mathfrak{L}'|_{\mathcal{C}_{bnd}}$.

We show $\mathrm{Gaif}(\mathfrak{L}|_{\mathcal{C}_{bnd}}) = \mathrm{Gaif}(\mathfrak{L}')$: We have that $\mathrm{Gaif}(\mathfrak{L}') = \mathrm{Gaif}(\mathfrak{L}'|_{\mathcal{R}})$ because of $\mathfrak{L}' \models \psi_{gaifmann}$ and $\mathfrak{L}' \models \psi_{subgraph}$. We have $\mathrm{Gaif}(\mathfrak{L}|_{\mathcal{C}_{bnd}}) = \mathrm{Gaif}(\mathfrak{L}|_{\mathcal{R}})$ because of $\mathfrak{L} \models \psi_{gaifmann}$. We have $\mathrm{Gaif}(\mathfrak{L}'|_{\mathcal{R}}) = \mathrm{Gaif}(\mathfrak{L}|_{\mathcal{R}})$ because of $rank(\mathfrak{L},\mathfrak{L}') = 0$. Thus, the claim follows. ◀

Finally, we show that there always is a sequence of models $\mathfrak{L}_0,\ldots,\mathfrak{L}_p$ with $\mathfrak{L} = \mathfrak{L}_0$, $\mathfrak{L}_i \models \alpha^\circ$ for each $0 \le i \le p$ and $rank(\mathfrak{L}_p,\mathfrak{L}') = 0$. We obtain each $\mathfrak{L}_{i+1}$ from $\mathfrak{L}_i$ by swapping edges, which has the consequence that the universe does not change, every each node u keeps its 1-type and the set of realized 2-types remains invariant throughout the construction of the sequence.

▶ **Lemma 4.8.** *There is a sequence of $\mathcal{D}_{unb}$-structures $\mathfrak{L}_0,\ldots,\mathfrak{L}_p$, with universe M and $\mathfrak{L} = \mathfrak{L}_0$, such that:*

(1) *$1\text{-}\mathfrak{tp}^{\mathfrak{L}_i}(u) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}'}(u)$ for all $u \in M$,*
(2) *$2\text{-}\mathfrak{tp}(\mathfrak{L}_i|_{\mathcal{E}}) = 2\text{-}\mathfrak{tp}(\mathfrak{L}_{i+1}|_{\mathcal{E}})$ for all $0 \le i < p$,*
(3) *$\mathfrak{L}_i \models \alpha^\circ$, (in particular $\mathfrak{L}_i$ is $\mathcal{T}$-functional and chromatic),*
(4) *$rank(\mathfrak{L}_i,\mathfrak{L}') > rank(\mathfrak{L}_{i+1},\mathfrak{L}')$ for all $0 \le i < p$, and $rank(\mathfrak{L}_p,\mathfrak{L}') = 0$.*

Proof. Assume we have already defined $\mathfrak{L}_i$ and $rank(\mathfrak{L}_i,\mathfrak{L}') > 0$. In the following we will define $\mathfrak{L}_{i+1}$. Because of $rank(\mathfrak{L}_i,\mathfrak{L}') > 0$ we can choose some elements $u,v,w \in M$ and $j \in [k]$ such that $\mathfrak{L}_i \models R_j(u,v)$, $\mathfrak{L}' \models R_j(u,w)$ and $v \neq w$. Let $\lambda = 2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(u,v)$. We have $\lambda \in \mathcal{T}\text{-MsgTypes}(\mathcal{E})$ because of $R_j \in \mathcal{R}$. We have $\mathfrak{L}_i \models P_\lambda(u)$ because of $\mathfrak{L}_i \models \psi_{typesUnb}$. Because $1\text{-}\mathfrak{tp}(u)^{\mathfrak{L}_i} = 1\text{-}\mathfrak{tp}(u)^{\mathfrak{L}'}$ we have $\mathfrak{L}' \models P_\lambda(u)$. Thus, $\mathfrak{L}_i \models \mathfrak{char}_{\lambda_y}(v)$ and $\mathfrak{L}' \models \mathfrak{char}_{\lambda_y}(w)$. With $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i}(w) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}'}(w)$ we get $\mathfrak{L}_i \models \mathfrak{char}_{\lambda_y}(w)$, and thus $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(v) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(w)$. We proceed by a case distinction:

Case 1: λ^{-1} is a $\mathcal{T}$-message type

We have $\mathfrak{L}' \models P_{\lambda^{-1}}(w)$ because of $\mathfrak{L}' \models \psi_{typesBnd}$. We get $\mathfrak{L}_i \models P_{\lambda^{-1}}(w)$ because of $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i}(w) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}'}(w)$. With $\mathfrak{L}_i \models \psi_{typesUnb}$, there is an element $a \in M$ such that $\lambda = 2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(a,w)$ and $\mathfrak{L}_i \models P_\lambda(a)$. We note that $u \neq a$ because of $\mathfrak{L}_i \models R_j(a,w)$, $v \neq w$ and v is the unique element with $\mathfrak{L}_i \models R_j(u,v)$ (using $\mathfrak{L}_i \models \psi_{functionality}$). Moreover, $\mathfrak{L}_i \models \mathfrak{char}_{(\lambda^{-1})_y}(a)$ and $\mathfrak{L}' \models \mathfrak{char}_{(\lambda^{-1})_y}(u)$. With $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i}(u) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}'}(u)$ we get $\mathfrak{L}_i \models \mathfrak{char}_{(\lambda^{-1})_y}(u)$, and thus $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(u) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(a)$.

We note that the edges (u,w), (w,u), (a,v) and (v,a) do not have $\mathcal{T}$-message types (*) because $\mathfrak{L}_i$ is chromatic, $u \neq a$, $v \neq w$, $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(v) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(w)$, $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(u) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|\mathcal{E}}(a)$ and the edges (u,v), (v,u), (a,w) and (w,a) have $\mathcal{T}$-message types. Similarly, we get $\lambda \neq 2\text{-}\mathfrak{tp}^{\mathfrak{L}'|\mathcal{E}}(u,v)$ and $\lambda \neq 2\text{-}\mathfrak{tp}^{\mathfrak{L}'|\mathcal{E}}(a,w)$ (**) because $\mathfrak{L}'$ is chromatic, $\lambda = 2\text{-}\mathfrak{tp}^{\mathfrak{L}'|\mathcal{E}}(u,w)$ and λ and λ^{-1} are $\mathcal{T}$-message types.

We define $\mathfrak{L}_{i+1}$ as follows: The unary relations of $\mathfrak{L}_{i+1}$ are defined such that we have $1\text{-}\mathfrak{tp}(u)^{\mathfrak{L}_{i+1}} = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i}(u)$ for all elements $u \in M$. We obtain the binary relations of $\mathfrak{L}_{i+1}$ by swapping the edges (u,w) and (u,v) as well as (a,w) and (a,v) in $\mathfrak{L}_i$: we set $2\text{-}\mathfrak{tp}^{\mathfrak{L}_{i+1}|_\mathcal{E}}(u,w) = \lambda$, $2\text{-}\mathfrak{tp}^{\mathfrak{L}_{i+1}|_\mathcal{E}}(a,v) = \lambda$, $2\text{-}\mathfrak{tp}^{\mathfrak{L}_{i+1}|_\mathcal{E}}(a,w) = 2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(a,v)$ and $2\text{-}\mathfrak{tp}^{\mathfrak{L}_{i+1}|_\mathcal{E}}(u,v) = 2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(u,w)$; these 2-types are well-defined because of $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(v) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(w)$. All other 2-types in $\mathfrak{L}_{i+1}|_\mathcal{E}$ are the same as in $\mathfrak{L}_i|_\mathcal{E}$. This completes the definition of $\mathfrak{L}_{i+1}$.

We now argue that $\mathfrak{L}_{i+1}$ satisfies properties (1)-(4). Clearly, $\mathfrak{L}_{i+1}$ satisfies (1) by definition. Because we only swapped edges from $\mathfrak{L}_i$ to $\mathfrak{L}_{i+1}$ we have (2). Because we only swapped swapped edges from $\mathfrak{L}_i$ to $\mathfrak{L}_{i+1}$ we get $\mathfrak{L}_{i+1} \models \psi_{gaifmann}$ and $\mathfrak{L}_{i+1} \models \psi_{subgraph}$ from $\mathfrak{L}_i \models \psi_{gaifmann}$ and $\mathfrak{L}_i \models \psi_{subgraph}$. Because of (*) and $2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(u,v) = \lambda = 2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(a,w)$ we get $\mathfrak{L}_{i+1} \models \psi_{typesUnb}$ from $\mathfrak{L}_i \models \psi_{typesUnb}$, that $\mathfrak{L}_{i+1}$ is $\mathcal{T}$-functional because $\mathfrak{L}_i$ is $\mathcal{T}$-functional and $\mathfrak{L}_{i+1} \models \psi_{chromaticity}$ from $\mathfrak{L}_i \models \psi_{chromaticity}$ (using that $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(v) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(w)$ and $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(u) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(a)$). With $\mathfrak{L}_i \models \alpha^\circ$ and Lemma 4.1 we get $\mathfrak{L}_{i+1} \models \alpha^\circ$. Thus, $\mathfrak{L}_{i+1}$ satisfies property (3).

It remains to show property (4). Using (*), (**) and Lemma 4.6 we get $rank_u(\mathfrak{L}_{i+1}, \mathfrak{L}') < rank_u(\mathfrak{L}_i, \mathfrak{L}')$ and $rank_z(\mathfrak{L}_{i+1}, \mathfrak{L}') \leq rank_z(\mathfrak{L}_i, \mathfrak{L}')$ for $z \in \{v, w, a\}$. Moreover, $rank_z(\mathfrak{L}_{i+1}, \mathfrak{L}') = rank_z(\mathfrak{L}_i, \mathfrak{L}')$ for $z \in M \setminus \{u, v, w, a\}$. Thus, $rank(\mathfrak{L}_i, \mathfrak{L}') > rank(\mathfrak{L}_{i+1}, \mathfrak{L}')$.

Case 2: λ^{-1} is not a $\mathcal{T}$-message type

We note that the edge (w,u) does not have a $\mathcal{T}$-message type because $\mathfrak{L}_i$ is chromatic, $v \neq w$, $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(v) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(w)$ and the edge (u,v) has a $\mathcal{T}$-message type.

We define $\mathfrak{L}_{i+1}$ as follows: The unary relations of $\mathfrak{L}_{i+1}$ are defined such that we have $1\text{-}\mathfrak{tp}(u)^{\mathfrak{L}_{i+1}} = 1\text{-}\mathfrak{tp}(u)^{\mathfrak{L}_i}$ for all elements $u \in M$. We obtain the binary relations of $\mathfrak{L}_{i+1}$ by swapping edges in $\mathfrak{L}_i$: $2\text{-}\mathfrak{tp}^{\mathfrak{L}_{i+1}|_\mathcal{E}}(u,w) = \lambda$, and $2\text{-}\mathfrak{tp}^{\mathfrak{L}_{i+1}|_\mathcal{E}}(u,v) = 2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(u,w)$. These 2-types are well-defined because of $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(v) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(w)$. All other 2-types in $\mathfrak{L}_{i+1}|_\mathcal{E}$ are the same as in $\mathfrak{L}_i|_\mathcal{E}$. This completes the definition of $\mathfrak{L}_{i+1}$.

We now argue that $\mathfrak{L}_{i+1}$ satisfies properties (1)-(4). As in the previous case, one can argue that $\mathfrak{L}_{i+1}$ satisfies (1) and (2), $\mathfrak{L}_{i+1} \models \psi_{gaifmann}$ and $\mathfrak{L}_{i+1} \models \psi_{subgraph}$. Because (v,u) and (w,u) do not have $\mathcal{T}$-message types we get $\mathfrak{L}_{i+1} \models \psi_{typesUnb}$ from $\mathfrak{L}_i \models \psi_{typesUnb}$ and that $\mathfrak{L}_{i+1}$ is $\mathcal{T}$-functional because $\mathfrak{L}_i$ is $\mathcal{T}$-functional. We get $\mathfrak{L}_{i+1} \models \psi_{chromaticity}$ from $\mathfrak{L}_i \models \psi_{chromaticity}$, $\mathfrak{L}' \models \psi_{chromaticity}$ and $1\text{-}\mathfrak{tp}^{\mathfrak{L}_i}(w) = 1\text{-}\mathfrak{tp}^{\mathfrak{L}'}(w)$. With $\mathfrak{L}_i \models \alpha^\circ$ and Lemma 4.1 we get $\mathfrak{L}_{i+1} \models \alpha^\circ$. Thus, $\mathfrak{L}_{i+1}$ satisfies property (3).

It remains to show property (4). From Lemma 4.6 we get $2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(u,w) \neq \lambda$ and $2\text{-}\mathfrak{tp}^{\mathfrak{L}'|_\mathcal{E}}(u,v) \neq \lambda$. Again applying Lemma 4.6 we get $rank_u(\mathfrak{L}_{i+1}, \mathfrak{L}') < rank_u(\mathfrak{L}_i, \mathfrak{L}')$. Because $2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(v,u)$ and $2\text{-}\mathfrak{tp}^{\mathfrak{L}_i|_\mathcal{E}}(w,u)$ are not $\mathcal{T}$-message types, we get $rank_v(\mathfrak{L}_{i+1}, \mathfrak{L}') = rank_v(\mathfrak{L}_i, \mathfrak{L}')$ and $rank_w(\mathfrak{L}_{i+1}, \mathfrak{L}') = rank_w(\mathfrak{L}_i, \mathfrak{L}')$. Moreover, we have that $rank_z(\mathfrak{L}_{i+1}, \mathfrak{L}') = rank_z(\mathfrak{L}_i, \mathfrak{L}')$ for all $z \in M \setminus \{u, v, w\}$. Thus, $rank(\mathfrak{L}_i, \mathfrak{L}') > rank(\mathfrak{L}_{i+1}, \mathfrak{L}')$. ◀

5 From Bounded Tree-width to Binary Trees

This section is devoted to a discussion of the proof of Lemma 3.3. It is well-known that graphs of tree-width k can be encoded as trees whose vertices are labeled with a finite number of labels. The popular definition of tree-width based on tree decompositions can be used as the basis of such an encoding. Moreover, the class of graphs of tree-width k can be obtained as the image of a translation scheme on the class of trees [1]. A translation scheme is a tuple of formulas which induces an operation mapping structures to structures. Our proof of Lemma 3.3 relies on an encoding of structures of bounded tree-width into trees in terms of a translation scheme with some special properties.

A **translation scheme for $\mathcal{C}_2$ over $\mathcal{C}_1$** is a tuple $\mathbf{t} = \langle \phi, \psi_C : C \in \mathcal{C}_2 \rangle$ of MSO($\mathcal{C}_1$)-formulas such that ϕ has exactly one free first order variable and the number of free first order variables in each ψ_C is $arity(C)$. The formulas ϕ and ψ_C, $C \in \mathcal{C}_2$, do not have any free second order variables.[5] The **quantifier rank qr(t) of t** is the maximum of the quantifier ranks of ϕ and the ψ_C. $\mathbf{t}$ is **quantifier-free** if $\mathrm{qr}(\mathbf{t}) = 0$. The **induced transduction $\mathbf{t}^\star$** is a partial function from $\mathcal{C}_1$-structures to $\mathcal{C}_2$-structures which assigns a $\mathcal{C}_2$-structure $\mathbf{t}^\star(\mathfrak{A})$ to a $\mathcal{C}_1$-structure $\mathfrak{A}$ as follows. The universe of $\mathbf{t}^\star(\mathfrak{A})$ is $A_{\mathbf{t}} = \{a \in A : \mathfrak{A} \models \phi(a)\}$. The interpretation of $C \in \mathcal{C}_2$ in $\mathbf{t}^\star(\mathfrak{A})$ is $C^{\mathbf{t}^\star(\mathfrak{A})} = \left\{\bar{a} \in {A_{\mathbf{t}}}^{arity(C)} : \mathfrak{A} \models \psi_C(\bar{a})\right\}$. Due to the convention that structures do not have an empty universe, $\mathbf{t}^\star(\mathfrak{A})$ is defined iff $\mathfrak{A} \models \exists x \phi(x)$.

▶ **Example 5.1** (k-bounded structures)**.** Let $\mathcal{A}$ be a vocabulary. Recall from Section 4.2.2 that k-bounded $\mathcal{A}$-structures $\mathfrak{A}$ can be encoded in terms of $\mathcal{R}$-functional structures enriched with unary predicates encoding 2-types. For simplicity we assume $\mathcal{A}$ consists of binary relation symbols only. Let $\mathcal{R} = \{R_1, \ldots, R_k\}$ and let $\mathcal{X} = \{P_\lambda \mid \lambda \in \mathcal{R}\text{-MsgTypes}(\mathcal{A})\}$. Let $\mathbf{t}_{k-b.s.} = \langle \phi, \psi_C : C \in \mathcal{A} \rangle$ be the translation scheme for $\mathcal{A} \cup \mathcal{R}$ over $\mathcal{X} \cup \mathcal{R}$ given as follows: $\phi(x_0) = true$; for every $R_j \in \mathcal{R}$, $\psi_{R_j}(x_0, x_1) = R_j(x_0, x_1)$; and for every $C \in \mathcal{A}$, $\psi_C(x_0, x_1) = \bigvee_{i=1}^{k} (dir_{1,i} \vee dir_{2,i})$, where $dir_{j,i} = \bigvee_{\lambda \in \mathrm{Range}_j} R_i(x_0, x_1) \wedge P_\lambda(x_{j-1})$ for $j \in \{1, 2\}$. The inner disjunction $\bigvee_{\lambda \in \mathrm{Range}_1}$ in $dir_{1,i}$ ranges over message types $\lambda \in \mathcal{R}$-MsgTypes($\mathcal{A}$) such that both $C(x, y)$ and $R_i(x, y)$ belong to λ. The disjunction over Range_2 in $dir_{2,i}$ is similar except that here we require $C(y, x)$ and $R_i(x, y)$ to belong to λ.

The induced transduction $\mathbf{t}^\star_{k-b.s.}$ is the operation which takes encodings given as $(\mathcal{X} \cup \mathcal{R})$-structures to their corresponding k-bounded $(\mathcal{A} \cup \mathcal{R})$-structures. Given a $(\mathcal{X} \cup \mathcal{R})$-structure $\mathfrak{B}$, $\mathbf{t}^\star_{k-b.s.}(\mathfrak{B})$ has the same universe as $\mathfrak{B}$ since $\phi = true$. The reducts of $\mathbf{t}^\star_{k-b.s.}(\mathfrak{B})$ and $\mathfrak{B}$ to $\mathcal{R}$ are equal. There is a C-edge from an element a to an element b in $\mathbf{t}^\star_{k-b.s.}(\mathfrak{B})$ if there is $e \in \{(a, b), (b, a)\}$ whose 2-type is a $\mathcal{R}$-message type and the source of e is annotated with a predicate P_λ such that λ contains C in the relevant direction (i.e., $C(x, y) \in \lambda$ if the source of e is a, and $C(y, x) \in \lambda$ otherwise).

▶ **Lemma 5.2** (Fundamental property of translation schemes)**.** *Let* $\mathbf{t}$ *be a translation scheme for* $\mathcal{C}_2$ *over* $\mathcal{C}_1$*. There is a computable function* $\mathbf{t}^\sharp$ *from* MSO($\mathcal{C}_2$)*-sentences to* MSO($\mathcal{C}_1$)*-sentences such that for every* $\mathcal{C}_1$*-structure* $\mathfrak{A}$ *for which* $\mathbf{t}^\star(\mathfrak{A})$ *is defined and for every* MSO($\mathcal{C}_2$)*-sentence* θ*,* $\mathfrak{A} \models \mathbf{t}^\sharp(\theta)$ *if and only if* $\mathbf{t}^\star(\mathfrak{A}) \models \theta$*. We call* $\mathbf{t}^\sharp$ *the* **induced translation***.*

For an MSO($\mathcal{C}_2$)-sentence ζ, $\mathbf{t}^\sharp$ substitutes the relation symbols $C \in \mathcal{C}_2$ in ζ with the formulas ψ_C, requires that each of the free variables satisfies ϕ, and relativizes the quantification to ϕ. An inductive definition of $\mathbf{t}^\sharp$ is given in Definition 3.2 of [20]. It is not difficult to extend the inductive definition of $\mathbf{t}^\sharp$ with the counting quantifiers (See Appendix C).

▶ **Example 5.3.** Let $\mathcal{E}$ be a vocabulary consisting of a single binary relation symbol E. The translation scheme $\mathbf{t}_{sym} = \langle \phi, \psi_E \rangle$ for $\mathcal{E}$ over $\mathcal{E}$ is given by $\phi(x) = E(x, x)$ and $\phi_E(x, y) = E(x, y) \vee E(y, x)$. The induced transduction $\mathbf{t}^\star_{sym}$ maps an $\mathcal{E}$-structure $\mathfrak{C}$ to the symmetric closure of the substructure of $\mathfrak{C}$ consisting of elements with self-loops. For example, for the structure $\mathfrak{C}_n$ with universe $[n]$ in which E is interpreted as the natural linear order $\leq$ of $[n]$, $\mathbf{t}^\star_{sym}(\mathfrak{C}_n)$ is $\langle [n], [n] \times [n] \rangle$. Let $full = \forall x \forall y\, E(x, y)$. By Lemma 5.2, $\mathbf{t}^\star_{sym}(\mathfrak{C}_n) \models full$ iff $\mathfrak{C}_n \models \mathbf{t}^\sharp_{sym}(full) = \forall x\ (E(x, x) \to (\forall y\, (E(y, y) \to (E(x, y) \vee E(y, x)))))$.

[5] All translation schemes in this paper are scalar (i.e. non-vectorized). In the notation of [6], a translation scheme is a parameterless non-copying MSO-definition scheme with precondition formula $(x \approx x)$.

▶ **Lemma 5.4** (Quantifier-free translation schemes and C^2). *Let* $\mathbf{t}$ *be a quantifier-free translation scheme for* $\mathcal{C}_2$ *over* $\mathcal{C}_1$*. The induced translation* $\mathbf{t}^\sharp$ *maps* $C^2(\mathcal{C}_2)$*-formulas to* $C^2(\mathcal{C}_1)$*-formulas.*

For the proof of Lemma 3.3 we need a translation scheme for $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$-structures $\mathfrak{M}$ whose reducts $\mathfrak{M}|_{\mathcal{C}_{bnd}}$ have tree-width at most k. In order that α and β be mapped to an MSO$(\mathcal{D}_{bnd})$-sentence and a $C^2(\mathcal{C}_{unb})$-sentence respectively, we need that the translation scheme satisfy some additional properties.

▶ **Lemma 5.5.** *Let* $\mathcal{C}_{bnd}$ *and* $\mathcal{C}_{unb}$ *be vocabularies such that* $\mathcal{C}_{bnd} \cap \mathcal{C}_{unb}$ *only contains unary relation symbols. There exist the following effectively computable objects: (1) a vocabulary* $\mathcal{D}_{bnd}$ *consisting of the binary relation symbol* s *and unary relation symbols only, (2) a translation scheme* $\mathbf{tr} = \langle \phi, \psi_C : C \in \mathcal{C}_{bnd} \cup \mathcal{C}_{unb} \rangle$ *for* $\mathcal{C}_{bnd} \cup \mathcal{C}_{unb}$ *over* $\mathcal{D}_{bnd} \cup \mathcal{C}_{unb}$*, and (3) an* MSO$(\mathcal{D}_{bnd})$*-sentence* dom*, such that:*

(a) ϕ *is quantifier-free over* $\mathcal{D}_{bnd}$*,*

(b) *For every relation symbol* $C \in \mathcal{C}_{unb}$*,* ψ_C *is quantifier-free.*

(c) *For every relation symbol* $C \in \mathcal{C}_{bnd}$*,* ψ_C *is an* MSO$(\mathcal{D}_{bnd})$*-formula.*

(d) *Let* $\mathcal{K}$ *be the class of* $(\mathcal{D}_{bnd} \cup \mathcal{C}_{unb})$*-structures in which* s *is interpreted as a binary tree and which satisfy* dom*. The image of* $\mathcal{K}$ *under* $\mathbf{tr}^\star$ *is exactly the class of* $(\mathcal{C}_{bnd} \cup \mathcal{C}_{unb})$*-structures* $\mathfrak{M}$ *such that* $\mathfrak{M}|_{\mathcal{C}_{bnd}}$ *has tree-width at most* k*.*

We use the translation scheme from Example 5.1 to encode k-bounded structures using $\mathcal{R}$-functional structures. (Note that structures of tree-width k are k-bounded.) We encode $\mathcal{R}$-functional structures as annotated trees using the inductive construction of k-trees in Section 2. The proof of Lemma 5.5 bears technical similarities to the discussion in [7] of graphs of bounded tree-width.

We are now ready to prove Lemma 3.3. By Lemma 5.5(d) and Lemma 5.2, statement (i) in Lemma 3.3 holds iff there is a $(\mathcal{D}_{bnd} \cup \mathcal{C}_{unb})$-structure $\mathfrak{N}$ such that $s^{\mathfrak{N}}$ is a binary tree and $\mathfrak{N} \models dom \wedge \mathbf{tr}^\sharp(\alpha) \wedge \mathbf{tr}^\sharp(\beta)$. Let $\alpha' = dom \wedge \mathbf{tr}^\sharp(\alpha)$ and $\beta' = \mathbf{tr}^\sharp(\beta)$. By Lemma 5.5(c) and the definition of $\mathbf{tr}^\sharp$, $\alpha' \in \text{MSO}(\mathcal{D}_{bnd})$. Let $\mathbf{tr}|_{\mathcal{C}_{unb}}$ be the translation scheme for $\mathcal{C}_{unb}$ over $\mathcal{D}_{bnd} \cup \mathcal{C}_{unb}$ which agrees with $\mathbf{tr}$ on all formulas, i.e. $\mathbf{tr}|_{\mathcal{C}_{unb}} = \langle \phi, \psi_C : C \in \mathcal{C}_{unb} \rangle$. The image of $\mathbf{tr}|^\star_{\mathcal{C}_{unb}}$ on a structure $\mathfrak{M}$ is the $\mathcal{C}_{unb}$-reduct of the image of $\mathbf{tr}^\star$ on $\mathfrak{M}$. Since $\beta \in C^2(\mathcal{C}_{unb})$, ${\mathbf{tr}|_{\mathcal{C}_{unb}}}^\sharp(\beta)$ is well-defined and ${\mathbf{tr}|_{\mathcal{C}_{unb}}}^\sharp(\beta) = \beta'$. By Lemma 5.5(a,b), $\mathbf{tr}|_{\mathcal{C}_{unb}}$ is a quantifier-free translation scheme, implying that $\beta' \in C^2(\mathcal{D}_{bnd} \cup \mathcal{C}_{unb})$ by Lemma 5.4.

6 From MSO to C^2 on Binary Trees

The purpose of this section is to show that, on structures consisting only of a binary tree and additional unary relations, every MSO-sentence can be rewritten to a C^2-sentence which is equi-satisfiable and whose length is linear in the length of the input MSO-sentence. We start by introducing some tools from the literature.

▶ **Theorem 6.1** (Hintikka sentences). *Let* $\mathcal{C}$ *be a vocabulary. For every* $q \in \mathbb{N}$ *there is a finite set* $\mathcal{HIN}_{\mathcal{C},q}$ *of* MSO$(\mathcal{C})$ *sentences of quantifier rank* q *such that:*

1. *every* $\epsilon \in \mathcal{HIN}_{\mathcal{C},q}$ *has a model;*
2. *the conjunction of any two distinct sentences* $\epsilon_1, \epsilon_2 \in \mathcal{HIN}_{\mathcal{C},q}$ *is not satisfiable;*
3. *every* MSO$(\mathcal{C})$*-sentence* α *of quantifier rank at most* q *is equivalent to exactly one finite disjunction of sentences* $\mathcal{HIN}_{\mathcal{C},q}$*;*
4. *every* $\mathcal{C}$*-structure* $\mathfrak{A}$ *satisfies exactly one sentence* $\text{hin}_{\mathcal{C},q}(\mathfrak{A})$ *of* $\mathcal{HIN}_{\mathcal{C},q}$*.*

We may omit $\mathcal{C}$ *or* q *from the subscript when they are clear from the context.*

For a class of $\mathcal{C}$-structures $\mathcal{K}$ an n-ary operation Op over $\mathcal{C}$-structures is called *smooth over* $\mathcal{K}$, if for all $\mathfrak{A}_1, \ldots, \mathfrak{A}_n \in \mathcal{K}$, $\mathrm{hin}_{\mathcal{C},q}(Op(\mathfrak{A}_1, \ldots, \mathfrak{A}_n))$ depends only on $\mathrm{hin}_{\mathcal{C},q}(\mathfrak{A}_i)$: $i \in [n]$ and this dependence is computable[6]. We omit *"over $\mathcal{K}$"* when $\mathcal{K}$ consists of all $\mathcal{C}$-structures.

▶ **Theorem 6.2** (Smoothness).

1. *The disjoint union is smooth.*
2. *For every quantifier-free translation scheme* $\mathbf{t}$*, the operation* $\mathbf{t}^\star$ *is smooth.*
3. *Let* $\mathfrak{T}_1 \vec{\circ}\, \mathfrak{T}_2$ *denote the tree obtained by adding* $\mathfrak{T}_2$ *as the child of* $\mathfrak{T}_1$*. That is,* $\mathfrak{T}_1 \vec{\circ}\, \mathfrak{T}_2$ *is obtained from the disjoint union of* $\mathfrak{T}_1$ *and* $\mathfrak{T}_2$ *by adding an edge from the root of tree* $\mathfrak{T}_1$ *to the root of tree* $\mathfrak{T}_2$*. The operation* $\vec{\circ}$ *is smooth on labeled trees.*[7]

For an in-depth introduction to Hintikka sentences and smoothness and references to proofs see [14, Chapter 3, Theorem 3.3.2] and [20] respectively.

We are now ready for the main lemma of this section.

Let $\mathrm{rt}(x)$ be the sentence $\forall y\, \neg s(y, x)$. This sentence defines the root of the binary tree s.

▶ **Lemma 6.3.** *Let* $q \in \mathbb{N}$ *and let* $\mathcal{C}$ *be a vocabulary which consists only of the binary relation symbol* s *and (possibly) additional unary relation symbols. For every* $\epsilon \in \mathcal{HIN}_{\mathcal{C},q}$*, let* C_ϵ *be a new unary relation symbol. Let* $\mathrm{hin}(\mathcal{C}, q)$ *be the vocabulary which extends* $\mathcal{C}$ *with* $\{C_\epsilon : \epsilon \in \mathcal{HIN}_{\mathcal{C},q}\}$*. There is a computable* $C^2(\mathrm{hin}(\mathcal{C}, q))$*-sentence* $\Theta^{\mathrm{hin}}_{\mathcal{C},q}$*, and for every* $\omega \in \mathrm{MSO}(\mathcal{C})$ *with* $\mathrm{qr}(\omega) = q$*, there exists a computable* C^2*-sentence* ω_{hin} *such that:*

(i) *Every* $\mathcal{C}$*-structure* $\mathfrak{T}_0$ *in which* $s^{\mathfrak{T}_0}$ *is a binary tree has a unique expansion* $\mathfrak{T}_1$ *to the vocabulary* $\mathrm{hin}(\mathcal{C}, q)$ *satisfying* $\mathfrak{T}_1 \models \Theta^{\mathrm{hin}}_{\mathcal{C},q}$*.*

(ii) *Moreover, for* $\mathfrak{T}_0$ *and* $\mathfrak{T}_1$ *as in (i),* $\mathfrak{T}_0 \models \omega$ *iff* $\mathfrak{T}_1 \models \omega_{\mathrm{hin}}$*.*

The C^2*-sentence* ω_{hin} *is*

$$\forall x \left(\mathrm{rt}(x) \to \bigvee_{\epsilon \in \mathcal{HIN}_{\mathcal{C},q} :\, \epsilon \models \omega} C_\epsilon(x) \right)$$

The sentence $\Theta^{\mathrm{hin}}_{\mathcal{C},q}$ is defined so that for every $\mathfrak{T}_0$ there is a unique expansion $\mathfrak{T}_1$ such that $\mathfrak{T}_1 \models \Theta^{\mathrm{hin}}_{\mathcal{C},q}$. For every u in the universe T_1 of $\mathfrak{T}_1$, we will have $u \in C^{\mathfrak{T}_1}_\epsilon$ iff the subtree $\mathfrak{T}_u$ of $\mathfrak{T}_0$ whose root is u satisfies $\mathfrak{T}_u \models \epsilon$. Using the smoothness of $\vec{\circ}$, whether an element of $\mathfrak{T}_1$ belongs to $C^{\mathfrak{T}_1}_\epsilon$ depends only on its children. This can be axiomatized in C^2. Lemma 3.4 follows from Lemma 6.3 with $q = \mathrm{qr}(\alpha)$, $\mathcal{D} = \mathrm{hin}(\mathcal{C}, q)$, and $\omega_{\mathrm{hin}} = \gamma$.

Appendix B spells out the proof of Lemma 6.3.

7 MSO with Cardinality Constraints

$\mathrm{MSO}^{\mathrm{card}}$ denotes the extension of MSO with atomic formulas called *cardinality constraints* $\sum_{i=1}^{r} |X_i| < \sum_{i=1}^{t} |Y_i|$, where the X_i and Y_i are MSO variables, and $|X|$ denotes the cardinality of X. Let WS1S ($\mathrm{WS1S}^{\mathrm{card}}$) be the weak monadic second order theory (with cardinality constraints) of the structure $\langle \mathbb{N}, +1, < \rangle$. Let $\mathrm{MSO}^{\exists\mathrm{card}} \subseteq \mathrm{MSO}^{\mathrm{card}}$ be the set of sentences ρ such that (1) ρ is of the form $\rho = \exists X_1 \cdots \exists X_m \omega$, and (2) only the $X_1, \ldots, X_m$ participate in cardinality constraints.

▶ **Theorem 7.1.** *Given a sentence* $\rho \in \mathrm{MSO}^{\exists\mathrm{card}}$*, it is decidable*

---

[6] Smooth operations here are called *effectively smooth* in [20].

[7] The smoothness of $\vec{\circ}$ can be shown using the smoothness of the *fusion* operation, see the discussion in Sections 3 and 4 of [20].

(A) *whether* $\langle \mathbb{N}, +1, < \rangle \models \rho$, *and*

(B) *whether* ρ *is satisfiable by a finite structure of bounded tree-width.*

We present the proof idea here, and the proof of Theorem 7.1(B) below. The proof of Theorem 7.1(A) is in Appendix A.

The proof of Theorem 7.1 follows from Theorem 3.1. The main observation needed for (B) is that cardinality constraints can be expressed in terms of injective functions, which are axiomatizable in C^2. (A) is reducible to (B). The main observations for (A) are:

1. that $X_1, \ldots, X_m$ are contained in the substructure $\mathfrak{A}_1$ of $\langle \mathbb{N}, +1, < \rangle$ generated by $\{0, \ldots, \ell\}$ for some $\ell \in \mathbb{N}$,
2. that substructure $\mathfrak{A}_2$ of $\langle \mathbb{N}, +1, < \rangle$ generated by $\mathbb{N} - \{0, \ldots, \ell\}$ is isomorphic to $\langle \mathbb{N}, +1, < \rangle$, and therefore $\mathfrak{A}_2$ and $\langle \mathbb{N}, +1, < \rangle$ have the same weak monadic second order theory,
3. that the weak monadic second order theory of $\langle \mathbb{N}, +1, < \rangle$ is decidable,
4. and that $\langle \mathbb{N}, +1, < \rangle$ is a transduction $\mathbf{t}$ of $\mathfrak{A}_1 \sqcup \mathfrak{A}_2$.

It remains to use that the disjoint union $\sqcup$ is smooth.

Proof of Theorem 7.1(B). Let ρ be a $\mathrm{MSO}^{\exists\mathrm{card}}$ sentence, i.e. the outermost block of quantifiers in ρ is existential and only variables from the outermost blocks may appear in cardinality constraints. For simplicity we consider $\rho = \exists X_1 \exists X_2 \omega$ with only two quantifiers in the outermost block. W.l.o.g. the only cardinality constraint in ω is $|X_1| < |X_2|$. By a slight abuse of notation, we sometimes treat X_1 and X_2 as unary relation symbols. Let $\mathcal{C}_{bnd}$ extend the vocabulary of ρ with new unary relation symbols X_1,X_2,W_{img},W_{dom}. Let $\mathcal{C}_{unb}$ extend $\mathcal{C}_{bnd}$ with a new binary relation symbol B. Finite satisfiability of ω by a structure $\mathfrak{M}$ such that $\mathrm{tw}(\mathfrak{M}) \leq k$ can be reduced to finite satisfiability of a sentence $\alpha \wedge \beta$, $\alpha \in \mathrm{MSO}(\mathcal{C}_{bnd})$ and $\beta \in C^2(\mathcal{C}_{unb})$, by a structure $\mathfrak{A}_1$ such that $\mathrm{tw}(\mathfrak{A}_1|_{\mathcal{C}_{bnd}}) \leq k$. Let β be the C^2-sentence $\beta = (\mathrm{inj}_{12} \vee \mathrm{inj}_{21}) \wedge \mathrm{dom} \wedge \mathrm{img}$, where:

- inj_{12} expresses that B is an injective function from X_1 to X_2,
- inj_{21} expresses that B is an injective function from X_2 to X_1,
- dom expresses that the domain of B is W_{dom},
- and img expresses that the image of B is W_{img}.

For every $\mathcal{C}_{unb}$-structure $\mathfrak{A}_1$, $|X_1^{\mathfrak{A}_1}| < |X_2^{\mathfrak{A}_1}|$ iff $W_{\mathrm{dom}}^{\mathfrak{A}_1} = X_1^{\mathfrak{A}_1}$ and $X_2^{\mathfrak{A}_1} \backslash W_{\mathrm{img}}^{\mathfrak{A}_1} \neq \emptyset$. Let α be obtained from ω by substituting every $|X_1| < |X_2|$ by

$$\forall x\, (X_1(x) \leftrightarrow W_{\mathrm{dom}}(x)) \wedge \exists x\, (\neg W_{\mathrm{img}}(x) \wedge X_2(x))$$

For any $\mathcal{C}_{bnd}$-structure $\mathfrak{A}_0$ with $\mathrm{tw}(\mathfrak{A}_0) \leq k$, $\mathfrak{A}_0 \models \omega$ iff there is an expansion $\mathfrak{A}_1$ of $\mathfrak{A}_0$ such that $\mathfrak{A}_1 \models \alpha \wedge \beta$. The treatment of other cardinality constraints $\sum_{i=1}^{r} |X_i| < \sum_{i=1}^{t} |Y_i|$ is similar; it is helpful to assume w.l.o.g. that the sets X_i and Y_i are pairwise disjoint. ◀

▶ Remark. While we assumed for simplicity in (B) that X_1 and X_2 range over subsets of the universe, it is not hard to extend the proof to the case that X_1 and X_2 are guarded second order variables which range over subsets of any relation in the structure. This is true since we can use the translation scheme $\mathbf{tr}$ from Lemma 5.5 to obtain the structures of tree-width at most k as the image of $\mathbf{tr}^\star$ of labeled trees; X_1 and X_2 then translate naturally to monadic second order variables.

Acknowledgements. We are grateful to the referees for their detailed comments on the presentation of the paper.

References

1 S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12(2):308–340, 1991.

2 F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic handbook*. Cambridge University Press, 2003.

3 D. Calvanese, T. Kotek, M. Šimkus, H. Veith, and F. Zuleger. Shape and content. In *Integrated Formal Methods*, pages 3–17. Springer, 2014.

4 W. Charatonik and P. Witkowski. Two-variable logic with counting and trees. In *LICS*, pages 73–82. IEEE, 2013.

5 B. Courcelle. The monadic second-order logic of graphs. i. recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990.

6 B. Courcelle and J. Engelfriet. *Graph structure and monadic second-order logic: a language-theoretic approach*, volume 138. Cambridge University Press, 2012.

7 B. Courcelle and J. A. Makowsky. Fusion in relational structures and the verification of monadic second-order properties. *Mathematical Structures in Computer Science*, 12(02):203–235, 2002.

8 J. Flum and M. Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. Springer, 2006.

9 E. Grädel. On the restraining power of guards. *The Journal of Symbolic Logic*, 64(04):1719–1742, 1999.

10 E. Grädel and M. Otto. On logics with two variables. *Theoretical computer science*, 224(1):73–113, 1999.

11 E. Grädel, M. Otto, and E. Rosen. Two-variable logic with counting is decidable. In *LICS*, pages 306–317. IEEE, 1997.

12 Erich Grädel, Martin Otto, and Eric Rosen. Undecidability results on two-variable logics. In *STACS 97, 14th Annual Symposium on Theoretical Aspects of Computer Science, Lübeck, Germany, February 27 – March 1, 1997, Proceedings*, pages 249–260, 1997.

13 J.G. Henriksen, J. Jensen, M. Jørgensen, N. Klarlund, R. Paige, T. Rauhe, and A. Sandholm. *Mona: Monadic second-order logic in practice*. Springer, 1995.

14 W. Hodges. Model theory. In *Encyclopedia of Mathematics and its Applications*, volume 42. Cambridge University Press, 1993.

15 F. Klaedtke and H. Rueß. Monadic second-order logics with cardinalities. In *Automata, Languages and Programming*, pages 681–696. Springer, 2003.

16 T. Kotek and J. A. Makowsky. Connection matrices and the definability of graph parameters. *Logical Methods in Computer Science*, 10(4), 2014.

17 T. Kotek, M. Simkus, H. Veith, and F. Zuleger. Extending ALCQIO with trees. In *Logic in Computer Science LICS*, pages 511–522, 2015.

18 L. Lovász. *Large networks and graph limits*, volume 60. AMS, 2012.

19 P. Madhusudan, P. S. Thiagarajan, and Shaofa Yang. The MSO theory of connectedly communicating processes. In *FSTTCS*, pages 201–212, 2005.

20 J.A. Makowsky. Algorithmic uses of the Feferman–Vaught theorem. *Annals of Pure and Applied Logic*, 126(1):159–213, 2004.

21 G. Nelson and D.C. Oppen. Simplification by cooperating decision procedures. *TOPLAS*, 1(2):245–257, 1979.

22 A. Pnueli. The temporal logic of programs. In *FOCS*, pages 46–57. IEEE, 1977.

23 I. Pratt-Hartmann. Complexity of the two-variable fragment with counting quantifiers. *Journal of Logic, Language and Information*, 14(3):369–395, 2005.

24 A.J.A. Robinson and A. Voronkov. *Handbook of automated reasoning*. Elsevier, 2001.

25 D. Seese. The structure of the models of decidable monadic theories of graphs. *Annals of pure and applied logic*, 53(2):169–195, 1991.

26 W. Thomas. *Languages, automata, and logic.* Springer, 1997.
27 M.Y. Vardi. Why is modal logic so robustly decidable? *Descriptive complexity and finite models*, 31:149–184, 1996.

A Appendix: Proof of Theorem 7.1(A)

The unary function $+1$ is the *successor relation* of $\mathbb{N}$ and interprets the binary relation symbol suc. The binary relation $<$ is the *natural order relation* on $\mathbb{N}$. In the proof of (A) we will use the theory of Hintikka sentences as presented in Section 6 with one caveat, namely that instead of restricting to finite structures, we allow arbitrary structures. Theorems 6.1 and 6.2 hold for arbitrary structures. Theorems 6.1 guarantees the existence of a set $\mathcal{HIN}^{\mathrm{arb}}_{\mathcal{C},q}$ analogously to $\mathcal{HIN}_{\mathcal{C},q}$ for arbitrary structures. For every $\mathcal{C}$-structure $\mathfrak{A}$, Theorem 6.1 guarantees the existence of a sentence $\mathrm{hin}^{\mathrm{arb}}_{\mathcal{C},q}(\mathfrak{A})$ of $\mathcal{HIN}^{\mathrm{arb}}_{\mathcal{C},q}$ analogously to $\mathrm{hin}_{\mathcal{C},q}(\mathfrak{A})$. For the rest of the section, we omit the superscript arb to simplify notation.

Consider $\rho = \exists X_1 \exists X_2 \omega$ for the vocabulary $\mathcal{C}_{\mathbb{N}}$ of $\langle \mathbb{N}, +1, < \rangle$. Let $\alpha \in \mathrm{MSO}(\mathcal{C}_{bnd})$ and $\beta \in C^2(\mathcal{C}_{unb})$ be as discussed in the proof of (B) above. The following are equivalent:

1. $\langle \mathbb{N}, +1, < \rangle \models \rho$
2. There are *finite* unary relations U_1 and U_2 such that $\langle \mathbb{N}, +1, <, U_1, U_2 \rangle \models \omega$. U_1 and U_2 interpret X_1 and X_2 respectively.
3. There are *finite* unary relations U_1 and U_2 and an expansion $\mathfrak{A}$ of $\langle \mathbb{N}, +1, <, U_1, U_2 \rangle$ such that $\mathfrak{A} \models \alpha \wedge \beta$. $\mathfrak{A}$ expands $\langle \mathbb{N}, +1, <, U_1, U_2 \rangle$ with interpretations of the symbols in $\mathcal{D}_{unb} = \{B, W_{\mathrm{dom}}, W_{\mathrm{img}}\}$.
4. $\rho' = \exists X_1 \exists X_2 (\alpha \wedge \beta)$ is satisfiable by an expansion of $\langle \mathbb{N}, +1, < \rangle$ with interpretations for the symbols of $\mathcal{D}_{unb}$.

We have *1. iff 2.* and *3. iff 4.* by the semantics of $\exists X_1 \exists X_2$ in weak monadic second order logic. We have *2. iff 3.* similarly to the discussion of α and β in the proof of (A) above. The rest of the proof is devoted to proving that 4. is decidable.

Observe that by the definition of β, and in weak monadic second order X_1 and X_2 are quantified to be finite sets, B is axiomatized to be a function with finite domain, and W_{dom} and W_{img} are finite. Hence models $\mathfrak{A}$ of ρ' can be decomposed into a finite part containing $B^{\mathfrak{A}}$, and an infinite part isomorphic to an expansion of $\langle \mathbb{N}, +1, < \rangle$ in which the symbols of $\mathcal{D}_{unb}$ as are interpreted as empty sets. We will use a similar decomposition, but first we want to move from the structure $\langle \mathbb{N}, +1, < \rangle$ and its expansions to $\langle \mathbb{N}, +1 \rangle$ and its expansions. There is a translation scheme $\mathbf{t}_<$ such that for every structure $\mathfrak{A} = \langle \mathbb{N}, +1, B^{\mathfrak{A}}, W^{\mathfrak{A}}_{\mathrm{dom}}, W^{\mathfrak{A}}_{\mathrm{img}} \rangle$, $\mathbf{t}^\star_<(\langle \mathfrak{A} \rangle) = \langle \mathfrak{A}, < \rangle$, where $\langle \mathfrak{A}, < \rangle$ is the expansion of $\mathfrak{A}$ with $<$. This is true since $<$ is MSO definable from $+1$. We have that ρ' is satisfied by an expansion of $\langle \mathbb{N}, +1, < \rangle$ iff $\mathbf{t}^\sharp_<(\rho')$ is satisfied by the same expansion of $\langle \mathbb{N}, +1 \rangle$.

Now we turn to the decomposition of models of $\mathbf{t}^\sharp_<(\rho')$ into a finite part containing $B^{\mathfrak{A}}$, and an infinite part isomorphic to a $\mathcal{D}_{unb}$-expansion of $\langle \mathbb{N}, +1 \rangle$. Let $\mathcal{D}_{unb+} = \langle \mathcal{D}_{unb}, + \rangle$ and note that $\mathcal{D}_{unb+}$ is the vocabulary of $\mathbf{t}^\sharp_<(\rho')$. For every $\mathcal{D}_{unb+}$-expansion $\mathfrak{P}$ of $\langle \mathbb{N}, +1 \rangle$ and every $n \in \mathbb{N}$, let $\mathfrak{P}_{1,n}$ and $\mathfrak{P}_{n,\infty}$ be the substructures of $\mathfrak{P}$ generated by $[n]$ and $\mathbb{N} \backslash [n-1]$ respectively. There is a translation scheme $\mathbf{u}$ such that $\mathbf{u}^\star(\mathfrak{P}_{1,n} \sqcup \mathfrak{P}_{n+1,\infty}) = \mathfrak{P}$ if $B^{\mathfrak{P}} \subseteq [n] \times [n]$. $\mathbf{u}$ existentially quantifies the set $[n]$ (which is the only non-empty finite set closed under suc and its inverse), 1 and n (as the first and last elements of $[n]$) and $n+1$ (as the only element without a suc-predecessor except for 1) and adds the edge $(n, n+1)$ to suc. We have $\mathfrak{P} \models \mathbf{t}^\sharp_<(\rho')$ iff $\mathfrak{P}_{1,n} \sqcup \mathfrak{P}_{n+1,\infty} \models \mathbf{u}^\sharp(\mathbf{t}^\sharp_<(\rho'))$. Note that the vocabulary of $\mathbf{u}^\sharp(\mathbf{t}^\sharp_<(\rho'))$ is $\mathcal{D}_{unb+}$. Let q be the quantifier rank of $\mathbf{u}^\sharp(\mathbf{t}^\sharp_<(\rho'))$.

The Hintikka sentence $\mathrm{hin}(\mathfrak{P}_{n+1,\infty})$ of quantifier rank q of $\mathfrak{P}_{n+1,\infty}$ is uniquely defined since $\mathfrak{P}_{n+1,\infty}$ is isomorphic to the expansion of $\langle \mathbb{N}, +1\rangle$ with empty sets. Moreover, $\mathrm{hin}(\mathfrak{P}_{n+1,\infty})$ is computable using that the theory of $\langle \mathbb{N}, +1\rangle$ is decidable. Hence, by the smoothness of the disjoint union, for every Hintikka sentence $\epsilon \in \mathcal{HIN}_{\mathcal{D}_{unb+},q}$ there is a computable set $E_\epsilon \subseteq \mathcal{HIN}_{\mathcal{D}_{unb+},q}$ such that $\mathrm{hin}(\mathfrak{P}_{1,n} \sqcup \mathfrak{P}_{n+1,\infty}) = \epsilon$ iff $\mathrm{hin}(P_{1,n}) \in E_\epsilon$. Then $\mathfrak{P}_{1,n} \sqcup \mathfrak{P}_{n+1,\infty} \models \mathbf{u}^\sharp(\mathbf{t}^\sharp_<(\rho'))$ iff $\mathfrak{P}_{1,n}$ satisfies the sentence $\bigvee_{(\epsilon,\epsilon')} \epsilon'$, where the $\bigvee_{(\epsilon,\epsilon')}$ ranges over pairs (ϵ, ϵ') such that (1) $\epsilon \in \mathcal{HIN}_{\mathcal{D}_{unb+},q}$, (2) $\epsilon \models \mathbf{u}^\sharp(\mathbf{t}^\sharp_<(\rho'))$, and (3) $\epsilon' \in E_\epsilon$. Hence, ρ' is satisfiable by an expansion of $\langle \mathbb{N}, +1, <\rangle$ iff $\bigvee_{(\epsilon,\epsilon')} \epsilon'$ is satisfiable by a finite structure in which suc is interpreted as a successor relation (i.e., as a simple directed path on the whole universe). Let suc-rel be the weak MSO-sentence such that the interpretation of suc is a successor relation. By Theorem 7.1(B), it is decidable whether, $\bigvee_{(\epsilon,\epsilon')} \epsilon' \wedge \text{suc-rel}$ is finitely satisfiable using that the class of simple directed paths annotated with unary relations has tree-width 1.

B Appendix: Proof of Lemma 6.3

For a leaf b, $\mathrm{hin}_q(\mathfrak{T}_b)$ depends only on the unary relations which b satisfies. By Theorem 6.2, for a vertex b with one child b_0 (two children b_0, b_1), $\mathrm{hin}_q(\mathfrak{T}_b)$ depends only on the unary relations which b satisfies and on $\mathrm{hin}_q(\mathfrak{T}_{b_0})$ (on $\mathrm{hin}_q(\mathfrak{T}_{b_0})$ and $\mathrm{hin}_q(\mathfrak{T}_{b_1})$).

Let

$$\Theta^{\mathrm{hin}}_{\mathcal{C},q} = \mathrm{part} \wedge \mathrm{leaves} \wedge \mathrm{ints}_1 \wedge \mathrm{ints}_2$$

where part says that $\{C_\epsilon : \epsilon \in \mathcal{HIN}_{\mathcal{C},q}\}$ partition the universe, and leaves, ints_1, and ints_2 define the C_ϵ for the leaves respectively the internal vertices of $\mathfrak{T}_0$ with one or two children.

We give part, leaves, ints_1 and ints_2 below. There are C^2 formulas $\mathrm{leaf}(x)$, $\mathrm{int}_1(x)$, and $\mathrm{int}_2(x)$, which express that x is a leaf, has one child, or has two children, respectively. Let

$$\mathrm{part} = \forall x \left(\left(\bigvee_{\epsilon \in \mathcal{HIN}_{\mathcal{C},q}} C_\epsilon(x) \right) \wedge \left(\bigwedge_{\epsilon_1 \neq \epsilon_2 \in \mathcal{HIN}_{\mathcal{C},q}} (\neg C_{\epsilon_1}(x) \vee \neg C_{\epsilon_2}(x)) \right) \right).$$

For $\mathcal{U} \subseteq \mathrm{un}(\mathcal{C})$, let $\mathfrak{O}_\mathcal{U}$ be a $\mathcal{C}$-structure with universe O of size 1 satisfying that $U^{\mathfrak{O}_\mathcal{U}} = \emptyset$ iff $U \in \mathcal{U}$, and $\mathrm{root}^{\mathfrak{O}_\mathcal{U}} = O$. Let

$$\begin{aligned}
\mathrm{ints}_1 &= \forall x \left(\mathrm{int}_1(x) \to \bigwedge_{(\mathcal{U},\epsilon_1,\epsilon_2) \in \mathrm{Range}_3} ((\mathrm{this}_\mathcal{U}(x) \wedge \mathrm{child}_{\epsilon_2}(x)) \to C_{\epsilon_1}(x)) \right) \\
\mathrm{this}_\mathcal{U}(x) &= \bigwedge_{U \in \mathrm{un}(\mathcal{C}), U \notin \mathcal{U}} U(x) \wedge \bigwedge_{U \in \mathrm{un}(\mathcal{C}), U \in \mathcal{U}} \neg U(x) \\
\mathrm{child}_{\epsilon_2}(y) &= \exists y\, s(x,y) \wedge C_{\epsilon_2}(y)
\end{aligned}$$

The conjunction $\bigwedge_{(\mathcal{U},\epsilon_1,\epsilon_2) \in \mathrm{Range}_3}$ ranges over tuples $(\mathcal{U}, \epsilon_1, \epsilon_2)$ such that $\mathcal{U} \subseteq \mathrm{un}(\mathcal{C})$, $\epsilon_1, \epsilon_2 \in \mathcal{HIN}_{\mathcal{C},q}$, and for every structure $\mathfrak{A} \in \mathcal{K}_{\mathrm{root}}$ with $\mathrm{hin}(\mathfrak{A}) = \epsilon_2$, $\mathrm{hin}(\mathfrak{O}_\mathcal{U} \overset{\to}{\circ} \mathfrak{A}) = \epsilon_1$.

We define:

$$\mathrm{leaves} = \forall x \left(\mathrm{leaf}(x) \to \bigwedge_{(\mathcal{U},\epsilon_1) \in \mathrm{Range}_4} \mathrm{this}_\mathcal{U}(x) \to C_{\epsilon_1}(x) \right)$$

The conjunction $\bigwedge_{(\mathcal{U},\epsilon_1)\in \mathrm{Range}_4}$ ranges over tuples $(\mathcal{U},\epsilon_1)$ such that $\mathcal{U} \subseteq \mathrm{un}(\mathcal{C})$, $\epsilon_1 \in \mathcal{HIN}_{\mathcal{C},q}$, and $\mathrm{hin}(\mathfrak{O}_{\mathcal{U}}) = \epsilon_1$. Finally, ints_2 is defined as follows:

$$\begin{aligned}
\mathrm{ints}_2 &= \forall x(\mathrm{int}_2(x) \to (\mathrm{intdist}(x) \wedge \mathrm{intsame}))\\
\mathrm{intdist}(x) &= \bigwedge_{(\mathcal{U},\epsilon_1,\epsilon_2,\epsilon_3)\in \mathrm{Range}_5} ((\mathrm{this}_{\mathcal{U}}(x) \wedge \mathrm{child}_{\epsilon_2}(x) \wedge \mathrm{child}_{\epsilon_3}(x)) \to C_{\epsilon_1}(x))\\
\mathrm{intsame}(x) &= \bigwedge_{(\mathcal{U},\epsilon_1,\epsilon_2)\in \mathrm{Range}_6} ((\mathrm{this}_{\mathcal{U}}(x) \wedge \mathrm{children}_{\epsilon_2}(x)) \to C_{\epsilon_1}(x))\\
\mathrm{children}_{\epsilon_2}(x) &= \exists^{=2} y\, (s(x,y) \wedge C_{\epsilon_2}(y))
\end{aligned}$$

The conjunction $\bigwedge_{(\mathcal{U},\epsilon_1,\epsilon_2,\epsilon_3)\in \mathrm{Range}_5}$ in $\mathrm{intdist}(x)$ ranges over $\mathcal{U} \subseteq \mathrm{un}(\mathcal{C})$ and $\epsilon_1,\epsilon_2,\epsilon_3 \in \mathcal{HIN}_{\mathcal{C},q}$ such that $\epsilon_2 \neq \epsilon_3$, and for every two structure $\mathfrak{A},\mathfrak{B} \in \mathcal{K}_{\mathrm{root}}$ with $\mathrm{hin}(\mathfrak{A}) = \epsilon_2$ and $\mathrm{hin}(\mathfrak{B}) = \epsilon_3$, $\mathrm{hin}((\mathfrak{O}_{\mathcal{U}} \overset{\to}{\circ} \mathfrak{A}) \overset{\to}{\circ} \mathfrak{B}) = \epsilon_1$. The conjunction $\bigwedge_{(\mathcal{U},\epsilon_1,\epsilon_2)\in \mathrm{Range}_6}$ in $\mathrm{intsame}(x)$ ranges over $\mathcal{U} \subseteq \mathrm{un}(\mathcal{C})$ and $\epsilon_1,\epsilon_2 \in \mathcal{HIN}_{\mathcal{C},q}$ such that for every structure $\mathfrak{A} \in \mathcal{K}_{\mathrm{root}}$ with $\mathrm{hin}(\mathfrak{A}) = \epsilon_2$, $\mathrm{hin}((\mathfrak{O}_{\mathcal{U}} \overset{\to}{\circ} \mathfrak{A}) \overset{\to}{\circ} \mathfrak{A}') = \epsilon_1$, where $\mathfrak{A}'$ is isomorphic to $\mathfrak{A}$ over a disjoint universe.

By definition, $\Theta^{\mathrm{hin}}_{\mathcal{C},q} \in C^2$. Let $\mathfrak{T}_0$ be a $\mathcal{C}$-structure in which s is interpreted as a binary tree. Let $\mathfrak{T}_1$ be the expansion of $\mathfrak{T}_0$ such that, for every element b of the universe $T_1 = T_0$ of $\mathfrak{T}_1$, $b \in C^{T_1}_{\mathrm{hin}(\mathfrak{T}_b)}$, where $\mathfrak{T}_b$ is the subtree of $\mathfrak{T}_0$ rooted at b. In particular, for the root r of $\mathfrak{T}_0$, $r \in C^{T_1}_{\mathrm{hin}(\mathfrak{T}_r)} = C^{T_1}_{\mathrm{hin}(\mathfrak{T}_0)}$. The structure $\mathfrak{T}_1$ is the unique expansion of $\mathfrak{T}_0$ such that $\mathfrak{T}_1 \models \Theta^{\mathrm{hin}}_{\mathcal{C},q}$, hence (i) holds. Using the definition of ω_{hin}, (ii) holds.

Note that $\Theta^{\mathrm{hin}}_{\mathcal{C},q}$ is a sentence in the description logic $\mathcal{ALCQIO}$ (which is a sublogic of C^2) discussed in [17]. The same is true for the sentences ω_{hin}.

C Appendix: The induced translation $\mathbf{t}^\sharp$

Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be vocabularies. Given a translation scheme $\mathbf{t} = \langle \phi, \psi_C : C \in \mathcal{C}_2\rangle$ for $\mathcal{C}_0$ over $\mathcal{C}_1$ we define the induced translation $\mathbf{t}^\sharp$ to be a function from MSO($\mathcal{C}_2$)-formulas to $\mathcal{C}_1$-formulas inductively as follows:

1. For $C \in \mathrm{un}(\mathcal{C}_2)$ or for monadic second order variables C, and for $\theta = C(x)$, we put

   $$\mathbf{t}^\sharp(\theta) = \psi_C(x) \wedge \phi(x)$$

2. For $C \in \mathrm{bin}(\mathcal{C}_2)$ and $\theta = C(x,y)$, we put $\mathbf{t}^\sharp(\theta) = \psi_C(x,y) \wedge \phi(x) \wedge \phi(y)$
3. For $x \approx y$, we put $\mathbf{t}^\sharp(\theta) = x \approx y \wedge \phi(x) \wedge \phi(y)$
4. For the Boolean connectives the translation distributes, i.e.
   - if $\theta = \theta_1 \vee \theta_2$ then $t^\sharp(\theta) = (\mathbf{t}^\sharp(\theta_1) \vee \mathbf{t}^\sharp(\theta_2))$
   - if $\theta = \neg\theta_1$ then $t^\sharp(\theta) = \neg\mathbf{t}^\sharp(\theta_1)$
5. For the existential quantifiers, we relativize to ϕ:
   If $\theta = Q\, y\, \theta_1$, where $Q \in \{\exists\} \cup \{\exists^{\leq n}, \exists^{\geq n}, \exists^{=n} : n \in \mathbb{N}\}$, we put $\mathbf{t}^\sharp(\theta) = Q\, y\, (\phi(y) \wedge \mathbf{t}^\sharp(\theta_1))$
   If $\theta = \exists U\, \theta_1$, we put $\mathbf{t}^\sharp(\theta) = \exists U\, (\mathbf{t}^\sharp(\theta_1) \wedge \forall y\, U(y) \to \phi(y))$

We have somewhat simplified the presentation in [20, Definition 2.3] to fit our setting. On the other hand, we have extended the presentation in [20] by the counting quantifiers. This is important for Lemma 5.4 on C^2-translations schemes. (Note that for the Lemma 5.4 it is not enough to use that counting quantifiers are definable in MSO.)