RWTH Aachen

Department of Computer Science
Technical Report

Axiomatising Tree-interpretable Structures

Achim Blumensath

ISSN 0935–3232 · Aachener Informatik Berichte · AIB-2001-10

RWTH Aachen · Department of Computer Science · December 2001

Axiomatising Tree-interpretable Structures

Achim Blumensath

RWTH Aachen

Abstract We introduce the class of tree-interpretable structures which generalises the notion of a prefix-recognisable graph to arbitrary relational structures. We prove that every tree-interpretable structure is finitely axiomatisable in guarded second-order logic with cardinality quantifiers.

1 Introduction

In recent years the investigation of algorithmic properties of *infinite* structures has become an established part of computer science. Its applications range from algorithmic group theory to databases and automatic verification. Infinite databases, for example, were introduced to model geometric and, in particular, geographical data (see [17] for an overview). In the field of automatic verification several classes of infinite transition systems and corresponding model-checking algorithms have been defined. For instance, model-checking for the modal μ-calculus over prefix-recognisable graphs is studied in [5], [18]. A further point of interest in this context is the bisimulation equivalence of such transition systems as considered in [24], [25].

Obviously, only restricted classes of infinite structures are suited for such an approach. In order to process a class $\mathcal{K}$ of infinite structures by algorithmic means two conditions must be met:

(i) Each structure $\mathfrak{A} \in \mathcal{K}$ must possess a *finite representation.*
(ii) The operations one would like to perform must be *effective* with regard to these representations.

One fundamental operation demanded by many applications is the evaluation of a query, that is, given a formula $\varphi(\bar{x})$ in some fixed logic and the representation of a structure $\mathfrak{A} \in \mathcal{K}$ one wants to compute a representation of the set $\varphi^{\mathfrak{A}} := \{\, \bar{a} \mid \mathfrak{A} \models \varphi(\bar{a}) \,\}$. Slightly simpler is the model-checking problem which asks whether $\mathfrak{A} \models \varphi(\bar{a})$ for some given $\bar{a}$. The class of tree-interpretable structures investigated in the present article has explicitly been defined in such a way that model-checking for MSO, monadic second order logic, is decidable. To the authors knowledge it is one of the largest natural classes with this property.

Several different notions of infinite graphs and structures have been considered in the literature:

– *Context-free graphs* [20], [21] are the configuration graphs of pushdown automata.

- *HR-equational graphs* [7] are defined by equations of hyperedge-replacement grammars.
- *Prefix-recognisable graphs* have been introduced in [6]. Several characterisations are presented in Section 3.
- *Automatic graphs* [16], [3], [4] are graphs whose edge relation is recognised by synchronous multihead automata.
- *Rational graphs* [16], [19] are graphs whose edge relation is recognised by asynchronous multihead automata.
- *Recursive graphs* [14] are graphs whose edge relation is recursive.

These classes of graphs form a strict hierarchy. Table 1 shows for which logic model-checking is still decidable for the various classes. $\mathrm{FO}(\exists^\kappa)$, $\mathrm{MSO}(\exists^\kappa)$, and $\mathrm{GSO}(\exists^\kappa)$ denote, respectively, first-order logic, monadic second-order logic, and guarded second-order logic extended by cardinality quantifiers. Σ_0 is the set of quantifier-free first-order formulae.

When investigating a class of finitely presented structures the question naturally arises which structures it contains. Usually it is quite simple to show that some structure belongs to the class by constructing a corresponding presentation. But the proof that such a presentation does not exists frequently requires more effort.

One possible approach consists in determining what additional information is needed in order to extract the presentation from a given structure. In the case of a tree-interpretable structure this information can be coded into a colouring of its elements and edges. A characterisation of these colourings amounts to one of the set of presentations of a structure. Besides determining whether a presentation exists such a characterisation can, for instance, be used to investigate the automorphism group of the structure.

In the present article we generalise the class of prefix-recognisable graphs to arbitrary relational structures and prove that each presentation corresponds to a $\mathrm{GSO}(\exists^k)$-definable colouring. This implies that each such structure is finitely axiomatisable in this logic. The outline of the article is as follows:

In Section 3 we review several characterisations of the class of prefix-recognisable graphs including characterisations in terms of languages, configuration graphs of pushdown automata, graph grammars, and interpretations.

Class	Logic
context-free	$\mathrm{GSO}(\exists^\kappa)$
HR-equational	$\mathrm{GSO}(\exists^\kappa)$
prefix-recognisable	$\mathrm{MSO}(\exists^\kappa)$
automatic	$\mathrm{FO}(\exists^\kappa)$
rational	Σ_0
recursive	Σ_0

Table 1. Decidability

The latter can be generalised to arbitrary relational structures most easily. The resulting class of tree-interpretable structures is defined in Section 4. After summarising some of its properties we also extend the characterisation via regular languages to this class.

Section 5 is devoted to the study of paths in tree-interpretable graphs. The presented results are mostly of a combinatorial nature and culminate in the proof that every connected component is spanned by paths with a certain property.

In Section 6 we prove our main theorem which states that all tree-interpretable structures are finitely axiomatisable in guarded second-order logic with cardinality quantifiers. We also show that the cardinality quantifiers are indeed needed.

Section 7 concludes the article with some lemmas about the orbits of the automorphism group of a tree-interpretable structure and the result that isomorphism is decidable for tree-interpretable structures of finite tree-width.

2 Preliminaries

Automata and trees. Let Σ be an alphabet. The complete tree over Σ is the structure $\mathfrak{T}_\Sigma := (\Sigma^*, (\mathrm{suc}_a)_{a\in\Sigma}, \preceq)$ where the suc_a denote the successor functions and $\preceq$ is the prefix-order. The longest common prefix of u and v is denoted by $u \sqcap v$. If $u = vw$ then we define $v^{-1}u := w$ and $uw^{-1} := v$.

For $u \in \Sigma^*$ and $k \in \mathbb{N}$ we write u/k for the prefix of u of length $|u| - k$, and $\mathrm{suf}_k\, u$ for the suffix of u of length k. In case $|u| < k$ we have $u/k = \varepsilon$ and $\mathrm{suf}_k\, u = u$. In particular, $(u/k)\,\mathrm{suf}_k\, u = u$ for all u and k.

Let $\leq_{\mathrm{lex}}$ be the lexicographic order and $\leq_{\mathrm{ll}}$ the length-lexicographic one defined by

$$\begin{aligned} x \leq_{\mathrm{lex}} y \quad &:\text{iff} \quad x \preceq y, \text{ or } wc \preceq x \text{ and } wd \preceq y \text{ for some} \\ &\qquad\qquad\qquad w \in \Sigma^*, c, d \in \Sigma \text{ with } c < d. \\ x \leq_{\mathrm{ll}} y \quad &:\text{iff} \quad |x| < |y|, \text{ or } |x| = |y| \text{ and } x \leq_{\mathrm{lex}} y. \end{aligned}$$

We denote automata by tuples $(Q, \Sigma, \Delta, q_0, F)$ with set of states Q, alphabet Σ, transition relation Δ, initial state q_0, and acceptance condition F.

Logic. Let us recall some basic definitions and fix our notation. Let $[n] := \{0, \dots, n-1\}$. We tacitly identify tuples $\bar{a} = a_0 \dots a_{n-1} \in A^n$ with functions $[n] \to A$ or with the set $\{a_0, \dots, a_{n-1}\}$. This allows us to write $\bar{a} \subseteq \bar{b}$ or $\bar{a} = \bar{b}|_I$ for $I \subseteq [n]$.

MSO, *monadic second-order logic*, extends first-order logic FO by quantification over sets. In *guarded second-order logic*, GSO, one can quantify over relations R of arbitrary arity with the restriction that every tuple $\bar{a} \in R$ is *guarded*, i.e., there is some relation S of the original structure that contains a tuple $\bar{b} \in S$ such that $\bar{a} \subseteq \bar{b}$. Note that every singleton a is guarded by $a = a$. For a more detailed definition see [15].

$\mathfrak{L}(\exists^\kappa)$ denotes the extension of the logic $\mathfrak{L}$ by *cardinality quantifiers* $\exists^\lambda$, for every cardinal λ, where $\exists^\lambda$ stands for "there are at least λ many".

A formula $\varphi(\bar{x})$ where each free variable is first-order defines on a given structure $\mathfrak{A}$ the relation $\varphi^{\mathfrak{A}} := \{\, \bar{a} \mid \mathfrak{A} \models \varphi(\bar{a}) \,\}$.

Definition 1. Let $\mathfrak{A} = (A, R_0, \dots, R_n)$ and $\mathfrak{B}$ be relational structures. A (one-dimensional) MSO-*interpretation* of $\mathfrak{A}$ in $\mathfrak{B}$ is a sequence

$$\mathcal{I} = \big\langle \delta(x),\ \varepsilon(x, y),\ \varphi_{R_0}(\bar{x}), \dots,\ \varphi_{R_n}(\bar{x}) \big\rangle$$

of MSO-formulae such that

$$\mathfrak{A} \cong \big(\delta^{\mathfrak{B}},\ \varphi^{\mathfrak{B}}_{R_0}, \dots,\ \varphi^{\mathfrak{B}}_{R_n}\big)/\varepsilon^{\mathfrak{B}}.$$

To make this expression well-defined we require that $\varepsilon^{\mathfrak{B}}$ is a congruence of the structure $\big(\delta^{\mathfrak{B}},\ \varphi^{\mathfrak{B}}_{R_0}, \dots,\ \varphi^{\mathfrak{B}}_{R_n}\big)$. We denote the fact that $\mathcal{I}$ is an MSO-interpretation of $\mathfrak{A}$ in $\mathfrak{B}$ by $\mathcal{I} : \mathfrak{A} \leq_{\mathrm{MSO}} \mathfrak{B}$ or $\mathfrak{A} = \mathcal{I}(\mathfrak{B})$.

The epimorphism $\big(\delta^{\mathfrak{B}},\ \varphi^{\mathfrak{B}}_{R_0}, \dots,\ \varphi^{\mathfrak{B}}_{R_n}\big) \to \mathfrak{A}$ is called *coordinate map* and also denoted by $\mathcal{I}$. If it is the identity function we say that $\mathfrak{A}$ is *definable* in $\mathfrak{B}$.

3 Prefix-recognisable graphs

Originally, the investigation of tree-interpretable structures was concerned only with transition systems. This subclass appears in the literature under several names using widely different definitions which all turned out to be equivalent. They are summarised in the next theorem. A more detailed description follows below.

Theorem 2. *Let $\mathfrak{G} = (V, (E_a)_{a \in A})$ be a graph. The following statements are equivalent:*

(1) $\mathfrak{G}$ *is prefix-recognisable.*
(2) $\mathfrak{G} = h^{-1}(\mathfrak{T}_2)|_C$ *for a rational substitution h and a regular language C.*
(3) $\mathfrak{G}$ *is the restriction to a regular set of the configuration graph of a pushdown automaton with ε-transitions.*
(4) $\mathfrak{G}$ *is* MSO-*interpretable in the binary tree $\mathfrak{T}_2$.*
(5) $\mathfrak{G}$ *is VR-equational.*

The equivalence of the first two items are due to Caucal [6], Stirling [25] mentioned the third characterisation, and Barthelmann [1] delivered the last two.

Definition 3. A graph is *prefix-recognisable* if it is isomorphic to a graph of the form $(S, (E_a)_{a \in A})$ where S is a regular language over some alphabet Σ and each E_a is a finite union of relations of the form

$$W(U \times V) := \{\, (wu, wv) \mid u \in U,\ v \in V,\ w \in W \,\}$$

for regular languages U, V, $W \subseteq \Sigma^*$.

Actually in the usual definition the reverse order $(U \times V)W$ is used. The above formulation was chosen as it fits better to the usual conventions regarding trees.

Example. The structure $(\omega, \mathrm{suc}, \leq)$ is prefix-recognisable. If we represent the universe by a^* the relations take the form

$$\mathrm{suc} = a^*(\varepsilon \times a) \quad \text{and} \quad \leq\, = a^*(\varepsilon \times a^*).$$

This representation can easily be generalised to one of the complete binary tree:

$$\mathrm{suc}_i = \{0,1\}^*(\varepsilon \times i) \quad \text{and} \quad \preceq\, = \{0,1\}^*(\varepsilon \times \{0,1\}^*).$$

Originally, Caucal defined the prefix-recognisable graphs in a different way. In order to obtain a class of graphs with decidable MSO-theory he defined two operations on graphs which preserve MSO-decidability and applied them to the binary tree $\mathfrak{T}_2$.

Definition 4. Let $\mathfrak{G} = (V,\ (E_a)_{a \in A})$ be a graph with universe $V \subseteq \{0,1\}^*$.

(1) The restriction $\mathfrak{G}|_C$ denotes the subgraph of $\mathfrak{G}$ induced by $C \subseteq \{0,1\}^*$.

(2) Let $\bar{A}$ be a disjoint copy of A and expand $\mathfrak{G}$ by the relations $E_{\bar{a}} := (E_a)^{-1}$ for $\bar{a} \in \bar{A}$. Given a set of labels B and a mapping h associating with every $b \in B$ a language $h(b) \subseteq (A \cup \bar{A})^*$, the inverse substitution $h^{-1}(\mathfrak{G})$ defines the graph $(V, (E'_b)_{b \in B})$ where E'_b consists of those pairs (u, v) such that in the expansion of $\mathfrak{G}$ there is a path from u to v labelled by some word in $h(b)$.

Proposition 1 (Caucal [6])**.** *A graph $\mathfrak{G}$ is prefix-recognisable if and only if it is isomorphic to $h^{-1}(\mathfrak{T}_2)|_C$ for some regular language C and mapping h such that $h(a) \subseteq \{0, 1, \bar{0}, \bar{1}\}^*$ is regular for all a.*

Example. $(\omega, \mathrm{suc}, \leq)$ can be defined with $C := 1^*$ and $h(\mathrm{suc}) := 1$, $h(\leq) := 1^*$.

Similarly to the characterisation of context-free graphs as configuration graphs of pushdown automata one can describe the class of prefix-recognisable graphs via some model of automaton. To do so one considers pushdown automata with ε-transitions where each configuration has either no outgoing ε-transitions or no outgoing non-ε-transitions. Then the ε-transitions are "factored out" in the following way: one takes only those vertices without outgoing ε-transitions and adds an a-transition between two vertices iff in $\mathfrak{G}$ there is a path between them consisting of one a-transition followed by arbitrarily many ε-transitions.

Proposition 2 (Stirling [25])**.** *A graph $\mathfrak{G}$ is prefix-recognisable if and only if it is the restriction to a regular set of the configuration graph of a pushdown automaton with ε-transitions where the ε-transitions are factored out in the way describe above.*

Example. A pushdown automaton for $(\omega, \mathrm{suc}, \leq)$ has the following configuration graph:

$$\begin{tikzcd}
q_0\varepsilon \arrow[r, "\mathrm{suc}"] \arrow[d, bend left, "\leq"] & q_0X \arrow[r, "\mathrm{suc}"] \arrow[d, bend left, "\leq"] & q_0XX \arrow[r, "\mathrm{suc}"] \arrow[d, bend left, "\leq"] & q_0XXX \arrow[r, "\mathrm{suc}"] \arrow[d, bend left, "\leq"] & \cdots \\
q_1\varepsilon \arrow[u, bend left, "\varepsilon"] \arrow[r, "\varepsilon"] & q_1X \arrow[u, bend left, "\varepsilon"] \arrow[r, "\varepsilon"] & q_1XX \arrow[u, bend left, "\varepsilon"] \arrow[r, "\varepsilon"] & q_1XXX \arrow[u, bend left, "\varepsilon"] \arrow[r, "\varepsilon"] & \cdots
\end{tikzcd}$$

Finally, one can characterise prefix-recognisable graphs via graph grammars. Using the notation of Courcelle [7], [9], [12] we consider the following operations on vertex-coloured graphs. Let C be a finite set of colours.

- $G + H$ is the disjoint union of G and H.
- $\varrho_\beta(G)$, for $\beta : C \to C$, changes the colour of the vertices from a to $\beta(a)$.
- $\eta^a_{b,c}(G)$ adds a-edges from each b-coloured vertex to all c-coloured ones.
- a denotes the graph with a single a-coloured vertex.

The clique-width of a graph $\mathfrak{G}$ is, by definition, the minimal number of colours one needs to write a term denoting $\mathfrak{G}$.

Definition 5. A countable coloured graph is *VR-equational* if it is the canonical solution of a finite system of equation of the form

$$x_0 = t_0, \quad \dots, \quad x_n = t_n$$

where the t_i are finite terms build up from the above operations. Further, we require that none of the t_i equals a single variable x_k.

Proposition 3 (Barthelmann [1])**.** *A graph is prefix-recognisable if and only if it is VR-equational.*

Since only finitely many colours can be used in a finite system of equations it follows that the clique-width of each VR-equational graph is finite.

Corollary 1. *Each prefix-recognisable graph is of finite clique-width.*

Example. If we colour the first element by a and the other ones by b we can define $(\omega, \mathrm{suc}, <)$ by

$$x_0 = \eta^<_{a,b}(x_1), \qquad x_1 = \varrho_{c\to b}\eta^{\mathrm{suc}}_{a,c}(a + x_2), \qquad x_2 = \varrho_{a\to c}(x_0).$$

4 Tree-interpretable structures

The characterisation of prefix-recognisable graphs in terms of interpretations is the one most easily generalised to arbitrary relational structures.

Definition 6. A structure $\mathfrak{A}$ is called *tree-interpretable* iff $\mathfrak{A} \leq_{\mathrm{MSO}} \mathfrak{T}_2$.

From this definition one can immediately deduce some basic properties of the class of tree-interpretable structures.

Proposition 4. *The class of tree-interpretable structures is closed under* MSO-*interpretations. In particular, it is closed under*

(1) *isomorphisms,*
(2) *definable expansions,*
(3) *expansion by finitely many constants,*
(4) *finite unions,*
(5) *factorisation by definable congruences, and*
(6) *substructures with definable universe.*

Proposition 5. MSO($\exists^\kappa$) *model checking is decidable for every tree-interpretable structure.*

Proof. MSO($\exists^\kappa$) model checking is decidable for $\mathfrak{T}_2$ and this property is conserved by MSO-interpretations. □

All tree-interpretable graphs are of finite clique-width. On the other hand, their tree-width can be unbounded as the example of the infinite clique $K_{\aleph_0}$ shows. A result of Courcelle [11] which was extended to tree-interpretable graphs by Barthelmann [2] shows that being of finite tree-width imposes a strong restriction on the structure of a tree-interpretable graph. Although stated only for graphs it also holds for arbitrary structures if one replaces $\mathfrak{G}$ by its Gaifman graph in (2)–(4).

Proposition 6 (Barthelmann [2], Courcelle [11])**.** *Let $\mathfrak{G}$ be a tree-interpretable graph. The following statements are equivalent:*

(1) $\mathfrak{G}$ *is HR-equational.*
(2) $\mathfrak{G}$ *has finite tree-width.*
(3) $\mathfrak{G}$ *does not contain the subgraph $K_{n,n}$ for some $n < \aleph_0$.*
(4) $\mathfrak{G}$ *is uniformly sparse.*

This characterisation allows us to extend Proposition 5 to GSO($\exists^\kappa$).

Theorem 7. *Let $\mathfrak{A}$ be a tree-interpretable structure.* GSO *model checking is decidable for $\mathfrak{A}$ if and only if $\mathfrak{A}$ is of finite tree-width. The same holds for* GSO($\exists^\kappa$).

Proof. If $\mathfrak{A}$ is of finite tree-width then GSO($\exists^\kappa$) collapses to MSO($\exists^\kappa$) which is decidable (see Courcelle [10], [13]).

The other direction is a special case of a result of Seese [23]. If $\mathfrak{A}$ has infinite tree-width then its Gaifman graph $\mathcal{G}(\mathfrak{A})$ contains subgraphs $K_{n,n}$ for all $n < \aleph_0$. Note that the grid $[n] \times [n]$ is bipartite since we can partition the vertices $(i, k) \in [n] \times [n]$ depending on whether $i + k$ is even or odd. Thus, $\mathcal{G}(\mathfrak{A})$ also contains subgraphs $[n] \times [n]$ for all $n < \aleph_0$. The result follows since GSO allows quantification over such subgraphs and the MSO theory of the class of grids is undecidable. □

Although the characterisation of tree-interpretable structures by interpretations is quite elegant, in actual proofs it is most of the time easier to work with a more concrete characterisation in terms of languages. Let us recall how automata are used to decide MTh($\mathfrak{T}_2$) (see [26] for an overview).

Definition 8. For sets $X_0, \dots, X_{n-1} \subseteq \{0,1\}^*$ let $T_{\bar{X}}$ be the $\mathscr{P}([n])$-labelled binary tree with $T(w) := \{\, i < n \mid w \in X_i \,\}$ for $w \in \{0,1\}^*$. For singletons $X_i = \{x_i\}$ we also write $T_{\bar{x}}$.

With this notation we can now state Rabin's famous tree theorem in the following way:

Theorem 9. *For each* MSO*-formula $\varphi(\bar{X}, \bar{x})$ there is a tree-automaton $\mathcal{A}$ that recognises the language $\{\, T_{\bar{X}\bar{x}} \mid \mathfrak{T}_2 \models \varphi(\bar{X}, \bar{x}) \,\}$.*

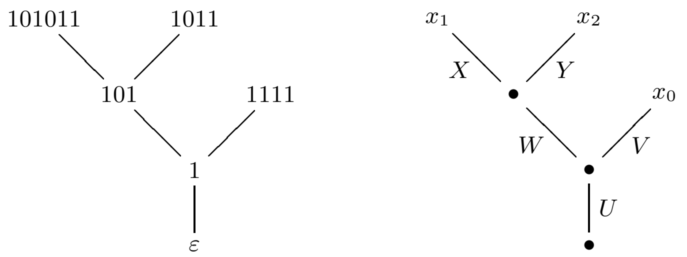

Figure 1. The branching structure of 1111, 1011, 101011 and its isomorphism type

Employing this correspondence we generalise the characterisation of prefix-recognisable graphs by relations of the form $W(U \times V)$ to arbitrary relational structures.

Definition 10. The *branching structure* of $x_0, \dots, x_{n-1} \in \Sigma^*$ is the partial order $(X, \preceq)$ where $X := \{\varepsilon\} \cup \{\, x_i \sqcap x_j \mid i,\ j < n \,\}$. The elements of X are called *branching points.*

Example. The branching structure of 1111, 1011, 101011 is depicted in Figure 1.

Note that for a fixed number of words there are only finitely many non-isomorphic branching structures.

Proposition 7. *An n-ary relation $R \subseteq (\{0,1\}^*)^n$ is* MSO*-definable in $\mathfrak{T}_2$ if and only if R is a finite union of relations R_i of the following form:*

(a) *All tuples $\bar{x} \in R_i$ have the same branching structure (up to isomorphism).*
(b) *For every pair of adjacent branching points u, v there is a regular language $W_{u,v}$ such that $\bar{x} \in R_i$ iff for each such pair u, v the word $u^{-1}v$ belongs to $W_{u,v}$.*

Proof. ($\Leftarrow$) Clearly, each such relation is MSO-definable.

($\Rightarrow$) We show that, if R is MSO-definable, then the labels of paths between branching points are regular. For simplicity we assume that the relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$ is binary. Let $\mathcal{A} = (Q, \mathscr{P}(\{0,1\}), \Delta, q_0, \Omega)$ be the tree-automaton associated with the MSO-definition of R in $\mathfrak{T}_2$. Let $\mathrm{occ}(t)$ denote the set of labels which occur at some vertex of the tree t. Let $L(\mathcal{A}, q)$ be the set of trees accepted by $\mathcal{A}$ if the initial state is q. We classify the states of $\mathcal{A}$ according to the set of labels which appear in trees that are accepted from this state.

$$\begin{aligned}
Q_\emptyset &:= \bigl\{ q \in Q \bigm| \textstyle\bigcup \mathrm{occ}(t) = \emptyset \text{ for all } t \in L(\mathcal{A}, q) \bigr\} \\
Q_0 &:= \bigl\{ q \in Q \bigm| \textstyle\bigcup \mathrm{occ}(t) = \{0\} \text{ for all } t \in L(\mathcal{A}, q) \bigr\} \\
Q_1 &:= \bigl\{ q \in Q \bigm| \textstyle\bigcup \mathrm{occ}(t) = \{1\} \text{ for all } t \in L(\mathcal{A}, q) \bigr\} \\
Q_{0,1} &:= \bigl\{ q \in Q \bigm| \textstyle\bigcup \mathrm{occ}(t) = \{0,1\} \text{ for all } t \in L(\mathcal{A}, q) \bigr\}
\end{aligned}$$

We construct languages U_q, V_q, and W_q such that $u \in U_q$, $v \in V_q$, and $w \in W_q$ if and only if there is an accepting run of $\mathcal{A}$ on $T_{\{wu\},\{wv\}}$ where the node w is labelled by q. Then

$$R = \bigcup_{q \in Q} W_q(U_q \times V_q).$$

We let W_q be the language recognised by the automaton $\bigl(Q, \{0,1\}, \Delta_{W_q}, q_0, \{q\}\bigr)$ where

$$\begin{aligned}\Delta_{W_q} := {} & \{\, (p, 0, p') \mid (p, \emptyset, p', p_0) \in \Delta,\ p_0 \in Q_\emptyset \,\} \\ & \cup \{\, (p, 1, p') \mid (p, \emptyset, p_0, p') \in \Delta,\ p_0 \in Q_\emptyset \,\}\end{aligned}$$

and let U_q be recognised by $\bigl(Q \mathbin{\dot\cup} \{q_f\}, \{0,1\}, \Delta_{U_q}, q, \{q_f\}\bigr)$ where

$$\begin{aligned}\Delta_{U_q} := {} & \{\, (p, 0, p') \mid (p, c, p', p_0) \in \Delta \text{ where } 0 \notin c \text{ and } p_0 \in Q_\emptyset \cup Q_1 \,\} \\ & \cup \{\, (p, 1, p') \mid (p, c, p_0, p') \in \Delta \text{ where } 0 \notin c \text{ and } p_0 \in Q_\emptyset \cup Q_1 \,\} \\ & \cup \{\, (p, c, q_f) \mid (p, c, p_0, p'_0) \in \Delta \text{ where } 0 \in c \text{ and } p_0, p'_0 \in Q_\emptyset \cup Q_1 \,\}\end{aligned}$$

V_q is defined analogously. □

Example. For the branching structure in Figure 1, a relation would be defined by five regular languages U, V, W, X, and Y with $R = U(V \times W(X \times Y))$.

Definition 11. Let $\mathfrak{A}$ be a tree-interpretable structure. Fixing an interpretation we can assume that the universe $A \subseteq \Sigma^*$ is regular and each relation R is specified by regular languages as in the preceding proposition. The *syntactic congruence* $\sim$ of $\mathfrak{A}$ (w.r.t. this interpretation) is the intersection of the syntactic congruences of all these languages. We denote the index of $\sim$ by I.

If some elements of a tree-interpretable structure are encoded by several words it becomes difficult to apply pumping arguments since the words obtained by pumping may encode the same element. Fortunately, for each tree-interpretable structure $\mathfrak{A}$ we can choose an interpretation where this does not happen.

Proposition 8. *If $\mathfrak{A} \leq_{\mathrm{MSO}} \mathfrak{T}_2$ then there is an interpretation $\mathcal{I} : \mathfrak{A} \leq_{\mathrm{MSO}} \mathfrak{T}_2$ where the coordinate map is injective.*

Proof. We prove for all regular languages $D \subseteq \{0,1\}^*$ and tree-interpretable equivalence relations $E \subseteq D \times D$ that there is a regular language $D' \subseteq D$ that contains exactly one element of each E-class. Then, the desired interpretation is obtained by replacing the formula defining the universe of $\mathfrak{A}$ by the one defining D'.

Denote the E-class of x by $[x]$, define $p_{[x]} := \inf_{\preceq} [x]$ and $s_x := (p_{[x]})^{-1}x$. Let $\varphi_p(x, y)$ be a MSO-definition of the function $x \mapsto p_{[x]}$. Finally, let s be the number of states of the automaton associated with E. We claim that each class $[x]$ contains an element of length less than $|p_{[x]}| + s$. Thus, one can define

$$D' := \{\, x \in D \mid s_x \leq_{\mathrm{ll}} s_y \text{ for all } y \in [x] \,\}$$

where the length lexicographic ordering $\leq_{\mathrm{ll}}$ is definable since the length of the words is bounded so that we only need to consider finitely many cases.

To prove the claim choose $x_0, x_1 \in [x]$ such that $x_0 \sqcap x_1 = p_{[x]}$. Since $(x_0, x_1) \in E$ there are regular languages U, V, and W such that $x_0 = wu$, $x_1 = wv$ for $u \in U$, $v \in V$, and $w \in W$ with $w \subseteq p_{[x]}$. If $|wu| \geq |p_{[x]}| + s$ then, by a pumping argument, there exists some $u' \in U$ such that $|p_{[x]}| \leq |wu'| < |p_{[x]}| + s$. Hence, $(wu', x_1) \in E$ is an element of the desired length. $\square$

This result allows us to identify the elements of a tree-interpretable structure with the unique word encoding them. We will do so tacitly in the remainder of the article. We conclude this section with a combinatorial lemma whose proof is based on a pumping argument.

Lemma 1. *Let $\mathfrak{A}$ be a tree-interpretable structure and $\varphi(x, y) \in \mathrm{MSO}(\exists^\kappa)$ such that, for every $a \in A$, there are only finitely many elements $b \in A$ with $\mathfrak{A} \models \varphi(a, b)$. There is a constant k such that $\varphi(a, b)$ implies $b/k \prec a$. In particular, $|\varphi(a, A)| \in \mathcal{O}(|a|)$.*

Proof. Consider the syntactic congruence $\sim$ of the expansion $(\mathfrak{A}, \varphi^{\mathfrak{A}})$ and let $k := I$ be its index. If there are elements a and b satisfying φ with $|b| \geq |a \sqcap b| + I$ then there are words $b/k \preceq x \prec y \preceq b$ with $x \sim y$. Let $u := x^{-1}y$ and $z := y^{-1}b$. Then $(a, b) \in \varphi^{\mathfrak{A}}$ implies that $(a, xu^i z) \in \varphi^{\mathfrak{A}}$ for all $i \in \mathbb{N}$. Contradiction. $\square$

5 Paths in tree-interpretable graphs

In this section we consider a fixed tree-interpretable graph $\mathfrak{G} = (V, E_0, \ldots, E_{r-1})$. By replacing each edge relation $E_a = \bigcup_i W_i(U_i \times V_i)$ by several relations $E_a^i := W_i(U_i \times V_i)$ we may assume that $E_a = W_a(U_a \times V_a)$ for regular languages $U_a, V_a, W_a \subseteq \Sigma^*$. We also add the relation $E_{a^-} := (E_a)^{-1}$ for each edge relation E_a. Note that these operations do not affect the syntactic congruence $\sim$.

Remark 1. By Proposition 7 we can choose U_a, V_a, W_a such that

$$(x, y) \in W_a(U_a \times V_a)$$
$$\text{iff} \quad x \sqcap y \in W_a, \ (x \sqcap y)^{-1}x \in U_a, \text{ and } (x \sqcap y)^{-1}y \in V_a.$$

Definition 12. The *base-point* of an edge $(a, b) \in W(U \times V)$ is the longest word w contained in W such that $w^{-1}a \in U$ and $w^{-1}b \in V$. The *spine* of a path is the sequence of the base-points of its edges.

Definition 13.

(1) A *path above c* is a path $a_0, \ldots, a_n$ such that $c \preceq a_i$ for all i.
(2) A path $a_0, \ldots, a_n$ is *bounded by l* if $|a_i| \leq l$ for all i.
(3) A sequence $a_0, \ldots, a_n$ is *k-increasing* if $|a_j| \geq |a_i| - k$ for all $j > i$.
(4) A path $a_0, \ldots, a_n$ with spine $w_0, \ldots, w_{n-1}$ is called *k-normal* if the path and its spine are k-increasing and $a_i/k \preceq a_j$ for all $i \leq j$.

The aim of this section is to show that every vertex can be reached by a k-normal path. The importance of such paths stems from the fact that, by following a k-normal path to a vertex x, one can compute certain information about x like its $\sim$-class. We start with some immediate observations.

Lemma 2. *Let $a_0, \dots, a_n$ be a path with spine $w_0, \dots, w_{n-1}$.*

(1) *Either $w_i \preceq w_{i+1}$ or $w_i \succeq w_{i+1}$ for all $i < n-1$.*
(2) *If $w_0, \dots, w_{n-1}$ is k-increasing then $w_i/k \preceq w_j$ for all $i < j$.*

The next two lemmas can be used to find a k-normal path once we have shown how to obtain a path with k-increasing spine.

Lemma 3. *Let a_0, a_1, a_2 be a path with spine w_0, w_1. There exists a vertex a_1' of length $|a_1'| < \max\{|w_0|, |w_1|\} + I$ such that a_0, a_1', a_2 is a path with spine w_0, w_1.*

Proof. W.l.o.g. assume that $w_1 \preceq w_0$. Suppose that $|a_1| \geq |w_0| + I$. Since $a_1 \succeq w_0$ there are prefixes $w_0 \preceq x \prec y \preceq a_1$ such that $(w_0)^{-1}x \sim (w_0)^{-1}y$. Setting $a_1' := x(y^{-1}a_1)$ we obtain a path a_0, a_1', a_2 with $|a_1'| < |a_1|$. By iteration, if necessary, we obtain a vertex of the desired length. □

Lemma 4. *Let $w_0, \dots, w_{n-1}$ be a k-increasing spine of some path from x to y. There exists a path $a_0, \dots, a_n$ with the same spine from x to y such that*

$$a_i/(k + I - 1) \preceq w_j \qquad \text{for all } 0 < i \leq j < n.$$

Proof. By the preceding lemma we can replace each a_i by some a_i' with $|a_i'| < \max\{|w_{i-1}|, |w_i|\} + I$, for $0 < i < n$. Since $w_{i-1}/k \preceq w_i$ it follows that $a_i'/(k + I - 1) \preceq w_{i-1}/k \preceq w_j$ for all $j \geq i$. □

In the proofs below we frequently need to remove parts of a path and glue the remaining pieces together. The following construction is the main tool in this process.

Definition 14. Let $a_0, \dots, a_n$ be a path with spine $w_0, \dots, w_{n-1}$. Let x and y be words such that $x \preceq w_i$ for all $i < n$, that is, there are words $u_0, \dots, u_n$, $v_0, \dots, v_{n-1}$ such that

$$a_i = xu_i \quad \text{and} \quad w_i = xv_i.$$

Shifting the path from x to y yields the sequences $a_0', \dots, a_n'$ and $w_0', \dots, w_{n-1}'$ where

$$a_i' := yu_i \quad \text{and} \quad w_i' := yv_i.$$

Lemma 5. *Using the same notation as in the preceding definition, $x \sim y$ implies that $a_0', \dots, a_n'$ is a path with spine $w_0', \dots, w_{n-1}'$.*

Proof. Since we have $w_i' \sim w_i$ and $(w_i')^{-1}a_i' = w_i^{-1}a_i$ and $(w_i')^{-1}a_{i+1}' = w_i^{-1}a_{i+1}$ it follows that $(a_i', a_{i+1}') \in E$ iff $(a_i, a_{i+1}) \in E$. □

Now we are ready to prove the main result needed to obtain k-normal paths.

Proposition 9. *Let $\mathfrak{G}$ be a tree-interpretable graph. There is a constant K_0 such that for all paths $a_0, \ldots, a_n$ with spine $w_0, \ldots, w_{n-1}$ there exists a path of length $m \leq n$ from a_0 to a_n with spine $w'_0, \ldots, w'_m$ where $|w'_i| < \max\{|w_0|, |w_{n-1}|\} + K_0$ for all $i \leq m$.*

Proof. We proceed in several steps.

Claim 1. *If $|w_i| \geq \max\{|w_0|,\ |w_{n-1}|\} + I$ for all $0 < i < n-1$, then there exists a path $a'_0, \ldots, a'_n$ from a_0 to a_n with $|a'_i| < |a_i|$ for all $0 < i < n$.*

The prerequisites imply that $w_0,\ w_{n-1} \prec w_i$ for all $0 < i < n-1$. Hence, either $w_0 \preceq w_{n-1}$ or $w_{n-1} \preceq w_0$. W.l.o.g. assume the latter. There exists a word x of length I such that $w_0 x \preceq w_i$ for all $0 < i < n-1$. Since $|x| = I$ there are prefixes $y \prec z \preceq x$ with $y \sim z$. The desired path is obtained from shifting the subpath $a_1, \ldots, a_{n-1}$ from $w_0 z$ to $w_0 y$.

By this claim we may assume that, for each subpath $a_k, \ldots, a_l$, there exists some index $k < i < l$ such that $|w_i| < \max\{|w_k|, |w_{l-1}|\} + I$.

Claim 2. *If $|w_1| \geq |w_0| + rI^2$, then there is a path $a'_0, \ldots, a'_m$ from a_0 to a_n with $m < n$.*

Let $w_{i_0}, \ldots, w_{i_t}$ be the subsequence of base-points w_i such that

$$|w_k| > |w_i| \geq |w_0| \qquad \text{for all } 0 < k < i.$$

By assumption

$$|w_{i_{k+1}}| < |w_{i_k}| < |w_{i_{k+1}}| + I$$

for all $k < t-1$. Hence, $t \geq rI$ and there exists indices $k < l$ in $\{i_0, \ldots, i_t\}$ such that $w_k \sim w_l$ and $(a_k, a_{k+1}), (a_l, a_{l+1}) \in E_s$ for some s. Since $|w_k| > |w_l|$ there is some word x with $w_k = w_l x$ and $w_l x \preceq w_i$ for all $0 < i < k$. Let $(a'_i)_i$ be the path obtained from $(a_i)_i$ by shifting the subpath $a_1, \ldots, a_k$ from $w_l x$ to w_l and removing the subpath $a_{k+1}, \ldots, a_l$.

By Claim 2 we may further assume that $|w_{i+1}| - |w_i| < rI^2$ for all $i < n-1$. Define $K_0 := r^3 I^4 |\Sigma|^{rI^2}$. The 3rd claim concludes the proof.

Claim 3. *There exists a path $a'_0, \ldots, a'_m$ from a_0 to a_n with spine $w'_0, \ldots, w'_{m-1}$ such that*

$$|w'_i| < \max\{|a_0|, |a_n|\} + r^3 I^4 |\Sigma|^{rI^2}.$$

Fix some base-point w_k such that $|w_k|$ is maximal, and consider the subsequences $w_{i_0}, \ldots, w_{i_s}$ and $w_{j_0}, \ldots, w_{j_t}$ of base-points w_i, for $i < k$, and w_j, for $j > k$, such that

$$\begin{aligned} |w_i| &< |w_l| \qquad \text{for all } i < l \leq k, \\ |w_j| &< |w_l| \qquad \text{for all } k \leq l < j. \end{aligned}$$

Assume that $|w_k| \geq \max\{|a_0|, |a_n|\} + r^3 I^4 |\Sigma|^{rI^2}$. By assumption, this implies that

$$s \geq r^3 I^4 |\Sigma|^{rI^2} / (rI^2) = r^2 I^2 |\Sigma|^{rI^2}.$$

For $i \in \{i_0, \dots, i_s\}$ define $f(i) \in \{j_0, \dots, j_t\}$ such that $|w_{f(i)}| \geq |w_i|$ is minimal. We colour each $i \in \{i_0, \dots, i_s\}$ with the tuple

$$\chi(i) := \big([w_i], [w_{f(i)}], w_i^{-1} w_{f(i)}, l, l'\big)$$

where l and l' are the indices with $(a_i, a_{i+1}) \in E_l$ and $(a_{f(i)}, a_{f(i)+1}) \in E_{l'}$. (Note that $w_i \preceq w_l$ for all $i \leq l \leq f(i)$.) Since

$$|w_{f(i)}| < |w_i| + rI^2$$

there are less than $I^2 |\Sigma|^{rI^2} r^2$ different colours. Thus, there are indices $i, i' \in \{i_0, \dots, i_s\}$, $i < i'$, with $\chi(i) = \chi(i')$. Let $w_{i'} = w_i x$. Then $w_i x \preceq w_l$ for $i' \leq l < f(i')$ and the desired path is obtained from $a_0, \dots, a_n$ by removing the subpaths $a_{i+1}, \dots, a_{i'}$ and $a_{f(i')+1}, \dots, a_{f(i)}$ and by shifting the subpath $a_{i'+1}, \dots, a_{f(i')}$ from $w_i x$ to w_i. $\square$

Corollary 2. *Let $\mathfrak{G}$ be a tree-interpretable graph. All elements a, b in the same component of V are connected by a path bounded by* $\max\{|a|, |b|\} + K_0 + I$.

Proof. Let $a_0, \dots, a_n$ be a path from a to b whose spine $w_0, \dots, w_{n-1}$ satisfies

$$|w_i'| < \max\{|w_0|, |w_{n-1}|\} + K_0 \qquad \text{for all } i < n.$$

Applying Lemma 3 we obtain a path $a_0', \dots, a_n'$ from a to b with

$$|a_i'| \leq \max\{|w_{i-1}|, |w_i|\} + I < \max\{|w_0|, |w_{n-1}|\} + K_0 + I. \qquad \square$$

Finally, we are able to prove the existence of k-normal paths.

Proposition 10. *Let $\mathfrak{G}$ be a tree-interpretable graph. There is a constant K such that each connected component of $\mathfrak{G}$ contains a vertex v, called its* root*, such that there are K-normal paths from v to all other vertices of the component.*

Proof. Let $K := K_0 + I - 1$. Choose v such that $|v|$ is minimal. Let $a_0, \dots, a_n$ be a path from v to some other vertex, and let $w_0, \dots, w_{n-1}$ be its spine. We transform it into a path with K_0-increasing spine as follows. Suppose there are indices $i < j$ such that $|w_j| < |w_i| - K_0$. Let $k < i$ be the greatest index such that $|w_k| < |w_j|$. By Proposition 9 there is a path $b_0, \dots, b_m$ from a_k to a_j whose spine is bounded by $|w_j| + K_0$. By iterating this operation we obtain a path with K_0-increasing spine. Applying Lemma 4 we obtain a path $a_0', \dots, a_n'$ from v to a_n with $a_i'/K \preceq w_j \preceq a_j', a_{j+1}'$ for all $0 < i \leq j < n$. It remains to prove that $a_0'/K = v/K \preceq a_1'/K$. Since $|a_1'| \geq |v|$ it is sufficient to show $v/K \preceq a_1'$. Assume that $|v \sqcap a_1'| < |v| - K$. Then $|a_1'| \geq |v| \geq |v \sqcap a_1'| + K$. Thus, there exists some b with $v \sqcap a_1' \preceq b \preceq a_1'$ and $|b| < |v \sqcap a_1'| + I < |v|$ such that $(v \sqcap a_1')^{-1} b \sim (v \sqcap a_1')^{-1} a_1'$. Therefore, $(v, a_1') \in E$ implies $(v, b) \in E$. This is a contradiction since the connected component of v does not contain vertices of smaller length. $\square$

6 Axiomatisations

Equipped with the combinatorial lemmas of the previous section we can present the main result of this article. Each tree-interpretable structure $\mathfrak{A}$ is finitely GSO($\exists^\kappa$)-axiomatisable, i.e., there is a GSO($\exists^\kappa$)-sentence $\psi_{\mathfrak{A}}$ such that $\mathfrak{B} \models \psi_{\mathfrak{A}}$ if and only if $\mathfrak{B} \cong \mathfrak{A}$. Actually, we will prove the slightly stronger statement that, for each tree-interpretable structure, there is a colouring of the guarded tuples such that the coloured structure is MSO($\exists^\kappa$)-axiomatisable. That is, the axiom consists of a sequence of existential non-monadic second-order quantifiers followed by an MSO($\exists^\kappa$)-formula.

6.1 The congruence colouring

The axiomatisation uses colourings of elements and pairs of elements that are of the following form:

Definition 15. (a) Let $\approx \subseteq \Sigma^* \times \Sigma^*$ be a congruence of finite index and let $k \in \mathbb{N}$. The $(\approx, k)$-*congruence colouring* $\chi^k_\approx$ maps words $x \in \Sigma^*$ to the pair

$$\chi^k_\approx(x) := \big([x/k]_\approx,\ \mathrm{suf}_k\, x\big)$$

and pairs $(x, y) \in \Sigma^* \times \Sigma^*$ to

$$\chi^k_\approx(x, y) := \big(\chi^k_\approx(w^{-1}x),\ \chi^k_\approx(w^{-1}y)\big)$$

where $w := x \sqcap y$.

(b) A $(\approx', k')$-colouring χ' *refines* the $(\approx, k)$-colouring χ if $\approx' \subseteq \approx$ and $k' \geq k$. We denote this fact by $\chi' \geq \chi$. The *common refinement* of the $(\approx_0, k_0)$-colouring χ_0 and the $(\approx_1, k_1)$-colouring χ_1 is the $(\approx_0 \cap \approx_1,\ \max\{k_0, k_1\})$-colouring denoted by $\chi_0 \sqcup \chi_1$.

Definition 16. The χ-*expansion* $(\mathfrak{A}, \chi)$ of a structure $\mathfrak{A}$ expands $\mathfrak{A}$ by unary and binary relations for each colour class where the binary colour classes consists only of pairs (x, y) which are guarded.

The restriction to guarded pairs is essential since GSO allows only quantification over relations of this form. Below we frequently will need to obtain the value $\chi(x, y)$ for pairs (x, y) which are not guarded. These values must be computed explicitly from available data. This is where k-normal paths come into play.

Lemma 6. *Let $\chi_1 \geq \chi_0$.*

(1) *There exists a function f with $\chi_0 = f \circ \chi_1$.*
(2) *$(\mathfrak{A}, \chi_0)$ is* FO*-interpretable in $(\mathfrak{A}, \chi_1)$.*

Lemma 7. *Let $\mathfrak{A}$ be a tree-interpretable structure, $\approx$ a congruence of finite index, and k a constant. The $\chi^k_\approx$-expansion $(\mathfrak{A}, \chi^k_\approx)$ of $\mathfrak{A}$ is also tree-interpretable.*

Proof. It is sufficient to note that, since $\approx$ is of finite index, each $\approx$-class is a regular language. □

We say that a set P of vertices codes a path between x and y if every element of P except for x and y is connected to exactly two other elements in P whereas x and y are connected to exactly one. Clearly, not every path can be coded in this way. Fortunately, for our purposes it is sufficient that, if there is a k-normal path between two vertices, then we can obtain a codable k-normal path between them by removing some vertices.

Lemma 8. *Let $\mathfrak{G}$ be a graph, χ a $(\approx, k)$-congruence colouring, and c a colour of χ. There is an* MSO*-formula $\varphi_c(P, x, y)$ such that $(\mathfrak{G}, \chi) \models \varphi_c(P, x, y)$ if and only if P codes a k-normal path from x to y and $\chi((x \sqcap y)^{-1} y) = c$.*

Proof. We label the elements $z \in P$ by the $(\approx, k')$-colour of $(x/k)^{-1} z$ for some $k \le k' \le 2k$. Since $x/k \preceq y$ we can compute $(x \sqcap y)^{-1} y$ from $\chi(x)$ and the label of y. To decide whether a given labelling is correct note that, if (z, z') is an edge of the path and z is labelled $([u], w)$, then the label of z' consists of the suffix w' of x' of length $\min\{2k, |w| + |x'| - |x|\}$ and the $\approx$-class of $(x/k)^{-1} z' (w')^{-1}$ both of which can be calculated from the colour of z. Note that, since the path is k-normal, we can ensure that the length of the stored suffix is at least $2k - k = k$. $\square$

6.2 Forests

We start slowly by showing that forests are finitely axiomatisable. We regard forests as partial orders such that the elements below any given one form a finite linear order. For any partial order let

$$\downarrow x := \{\, z \in A \mid z < x \,\} \qquad \text{and} \qquad \uparrow x := \{\, z \in A \mid x < z \,\}.$$

Lemma 9. *Let $\mathfrak{T} := (T, \le)$ be a tree-interpretable forest and $\chi^I_\sim$ the $(\sim, I)$-congruence colouring. $\chi^I_\sim(x) = \chi^I_\sim(y)$ implies that $\uparrow x \cong \uparrow y$.*

Proof. For each $b \in T$, there are only finitely many $a \le b$. By Lemma 1 it follows that $a \le b$ implies $a/I \preceq b$. Hence, $\uparrow a \subseteq (a/I)\Sigma^*$, and the function $f : (x/I)\Sigma^* \to (y/I)\Sigma^*$ mapping $(x/I)z$ to $(y/I)z$ is the desired isomorphism. $\square$

Theorem 17. *If $\mathfrak{T} := (T, \le)$ is a tree-interpretable forest and $\chi \ge \chi^I_\sim$ then the structure $(\mathfrak{T}, \chi)$ is finitely* $\mathrm{FO}(\exists^\kappa)$*-axiomatisable.*

Proof. Let $T_0 \subseteq T$ be the set of minimal elements, and for $x \in T$ let $S(x)$ be the set of its immediate successors. For $X \subseteq T$ let $\nu(X)$ be the function which maps each colour c to the number of elements $x \in X$ coloured c.

We claim that a partial order $\mathfrak{X} := (X, \le, \chi')$ is isomorphic to $(\mathfrak{T}, \chi)$ if and only if

(1) $\le$ is a partial order such that the set $\downarrow x$ is either empty or a finite linear order for all $x \in X$,
(2) $\nu(X_0) = \nu(T_0)$ where $X_0 \subseteq X$ is the set of minimal elements, and

(3) $\nu(S(x)) = \nu(S(u))$ for all vertices $x \in X$ and $u \in T$ with $\chi'(x) = \chi(u)$.

Clearly, all these conditions can be expressed in $\mathrm{FO}(\exists^\kappa)$.

To prove the nontrivial direction we construct an isomorphism $h : X \to T$ for a given order $\mathfrak{X}$ which satisfies the above conditions. Note that (1) implies that $\mathfrak{X}$ is a forest. Let $\mathrm{ht}(x) := |{\downarrow}x|$. We construct h as the limit of partial isomorphisms h_i, $i < \omega$, where

$$h_i : \{\, x \in X \mid \mathrm{ht}(x) \leq i \,\} \to \{\, u \in T \mid \mathrm{ht}(u) \leq i \,\}.$$

($i = 0$) As $\nu(X_0) = \nu(T_0)$ there is a bijection $h_0 : X_0 \to T_0$ that preserves the colouring.

($i > 0$) For each $x \in X$ with $\mathrm{ht}(x) = i - 1$ we choose a colour preserving bijection $g_x : S(x) \to S(h_{i-1}x)$. Note that (3) ensures its existence. h_i is the extension of h_{i-1} by all the g_x.

Using the preceding lemma it is easy to show that h is well-defined and indeed an isomorphism. □

6.3 Partial-orders

The next step consists in extending the result to tree-interpretable partial orders $\mathfrak{A} := (A, \leq)$ for which there is a constant $n \in \mathbb{N}$ such that $x \leq y$ implies $x/n \preceq y/n$ for all $x, y \in A$. To do so we have to define a forest in $\mathfrak{A}$. When speaking of paths we always consider undirected paths in this section, i.e., we ignore the direction of the edges.

Definition 18. Let $x \sqsubseteq y$ iff $x/n \preceq y/n$ and there is an undirected $\leq$-path $z_0, \dots, z_m$ from x to y with $x/n \preceq z_i/n$ for all $i \leq m$. Further, define $x \equiv y$ iff $x \sqsubseteq y$ and $y \sqsubseteq x$.

Lemma 10. *$(A, \sqsubseteq)/{\equiv}$ is a forest.*

Proof. It is sufficient to show that ${\downarrow}[x]_\equiv$ is a linear order for all $[x]_\equiv \in A/{\equiv}$. Let $[y]_\equiv$, $[z]_\equiv \sqsubseteq [x]_\equiv$ with $y/n \preceq z/n$. We claim that $[y]_\equiv \sqsubseteq [z]_\equiv$.

By definition, there are undirected $\leq$-paths $y_0, \dots, y_l$ form y to x and $z_0, \dots, z_m$ from z to x such that $y/n \preceq y_i/n$ and $z/n \preceq z_i/n$ for all i. Hence, also $y/n \preceq z_i/n$, and the path $y_0, \dots, y_l, z_{m-1}, \dots, z_0$ leading from y to z witnesses that $y \sqsubseteq z$. □

Using the result of the previous section we first prove that $(A, \sqsubseteq, \chi)$ is axiomatisable by defining a suitable copy of $(A, \sqsubseteq)/{\equiv}$ in it.

Lemma 11. *The subset $A_0 \subseteq A$ which consists of the lexicographically minimal elements of each $\equiv$-class is* MSO*-definable in $(A, \sqsubseteq, \chi^n_\sim)$.*

Proof. Since $x \equiv y$ implies $x/n = y/n$, one can determine whether $x \leq_{\mathrm{lex}} y$ by looking at $\mathrm{suf}_n\, x$ and $\mathrm{suf}_n\, y$. This information is contained in the colouring $\chi^n_\sim$. □

Proposition 11. *There is a congruence colouring χ_0 such that $(A, \sqsubseteq, \chi)$ is finitely* $\mathrm{MSO}(\exists^\kappa)$*-axiomatisable for every $\chi \geq \chi_0$.*

Proof. Let $\mathfrak{B} := (A, \sqsubseteq)$, and let $\delta(x)$ be the formula defining A_0 in $(\mathfrak{B}, \chi)$. We set $\chi_0 := \chi^n_\sim \sqcup \chi^{I_\mathrm{t}}_{\sim_\mathrm{t}}$ where $\sim_\mathrm{t}$ is the syntactic congruence corresponding to $\mathfrak{B}/{\equiv}$ and I_t is its index.

A structure $(\mathfrak{X}, \chi') := (X, \sqsubseteq', \chi')$ is isomorphic to $(\mathfrak{B}, \chi)$ if and only if there is an isomorphism

$$f : (\mathfrak{X}/{\equiv'}, \chi') \to (\mathfrak{B}/{\equiv}, \chi)$$

such that $[x]_{\equiv'} \cong f[x]_\equiv$ for all $x \in X$ where $\equiv' := \sqsubseteq' \cap \sqsupseteq'$. This condition is equivalent to the following ones:

(1) $\delta^\mathfrak{X}$ contains exactly one element of each $\equiv'$-class of X.
(2) $(\delta^\mathfrak{X}, \sqsubseteq', \chi') \cong (\delta^\mathfrak{B}, \sqsubseteq, \chi)$.
(3) $[x]_{\equiv'} \cong [a]_\equiv$ for all $x \in X$ and $a \in A$ with $\chi'(x) = \chi(a)$.

(1) and (3) are easily expressed in MSO. (2) can be checked since $\chi \geq \chi^{I_\mathrm{t}}_{\sim_\mathrm{t}}$ and therefore the forest $(\delta^\mathfrak{B}, \sqsubseteq, \chi) \cong (\mathfrak{B}/{\equiv}, \chi)$ is $\mathrm{FO}(\exists^\kappa)$-axiomatisable. □

In order to transfer the axiomatisability result from $(A, \sqsubseteq)$ to $\mathfrak{A}$, we have to show that each of the structures is definable in the other one.

Lemma 12.

(a) *If $x \sqsubseteq y$ then there is a K-normal path $z_0, \dots, z_m$ from x to y with $x/n \preceq z_i/n$ for all i.*
(b) *If $x \equiv y$ then there exists an undirected $\leq$-path $z_0, \dots, z_m$ from x to y with $x/n \preceq z_i/n$ and $|z_i| \leq |x| + K$ for all $i \leq m$.*
(c) *The relation $\sqsubseteq$ is* MSO*-definable in $(\mathfrak{A}, \chi^K_\sim)$.*
(d) *$(A, \sqsubseteq, \chi)$ is* MSO*-definable in $(\mathfrak{A}, \chi)$ for all $\chi \geq \chi^K_\sim$.*

Proof. (a) is implied by Proposition 10, and (b) follows from Proposition 9 since x and y are connected above $x/n = y/n$.

(c) We have $x \sqsubseteq y$ iff there is a K-normal undirected path $z_0, \dots, z_m$ from x to y with $x/n \preceq z_i/n$ for all i. Thus, $x \sqsubseteq y$ iff there is a K-normal path P such that each initial segment P' of P leads to some vertex z with $|z| \geq |x|$. It follows from Lemma 8 that the condition $|z| > |x|$ can be expressed by an MSO-formula.

(d) By (c) it remains to define the colouring $\chi(x, y)$ for $x \sqsubseteq y$. This can by done by Lemma 8 since there exists a K-normal path from x to y. □

Lemma 13. *$(\mathfrak{A}, \chi)$ is* MSO*-definable in $(A, \sqsubseteq, \chi)$ for all colourings $\chi \geq \chi^n_\sim$.*

Proof. Since $\leq$ is tree-interpretable and $x \leq y$ implies $x/n \preceq y/n$, there are sets $U([w]) \subseteq \Sigma^{\leq I}$ and $V([w]) \subseteq \Sigma^*/{\sim}$ for every class $[w] \in \Sigma^*/{\sim}$ such that

$$x \leq y \quad \text{iff} \quad x/n \preceq y/n, \quad w^{-1}x \in U([w]), \quad \text{and} \quad [w^{-1}y] \in V([w]).$$

where $w := x \sqcap y$. Since $x \leq y$ implies $x \sqsubseteq y$, all of the above conditions can be expressed in MSO using $\chi(x)$, $\chi(y)$, and $\chi(x, y)$. The colouring of $(\mathfrak{A}, \chi)$ is definable for the same reason. □

Theorem 19. *Let $\mathfrak{A} := (A, \leq)$ be a tree-interpretable partial-order and let $n \in \mathbb{N}$ be a constant such that $x \leq y$ implies $x/n \preceq y/n$ for all $x, y \in A$. There is a congruence colouring χ_0 such that $(\mathfrak{A}, \chi)$ is finitely* MSO($\exists^\kappa$)*-axiomatisable for every $\chi \geq \chi_0$.*

Proof. Let χ_0' be the colouring of Proposition 11. We set $\chi_0 := \chi_0' \sqcup \chi_\sim^K \sqcup \chi_\sim^n$. Let $\mathcal{I}$ be the MSO-definition of $(A, \sqsubseteq, \chi)$ in $(\mathfrak{A}, \chi)$. By the preceding lemma, a structure $(\mathfrak{X}, \chi')$ is isomorphic to $(\mathfrak{A}, \chi)$ if and only if $\mathcal{I}(\mathfrak{X}, \chi') \cong \mathcal{I}(\mathfrak{A}, \chi)$. The claim follows since $\mathcal{I}(\mathfrak{A}, \chi)$ is MSO($\exists^\kappa$)-axiomatisable by Proposition 11. □

6.4 The general case

Finally, we consider an arbitrary tree-interpretable structure $\mathfrak{A}$. For the reduction to the previous case we define, as above, a partial order $\leq$ and show that the structures $(A, \leq)$ and $\mathfrak{A}$ are definable within each other.

Definition 20. Let $x \vdash y$ if $x/I \preceq y/I$ and the pair (x, y) is guarded. Let $\leq$ be the reflexive and transitive closure of $\vdash$.

Lemma 14. *$(A, \leq, \chi)$ is* MSO*-definable in $(\mathfrak{A}, \chi)$ for all colourings $\chi \geq \chi_\sim^I$.*

Proof. The relation $\vdash$ is FO-definable, since one can tell whether $x/I \preceq y/I$ holds by looking at $\chi_\sim^I(x, y)$. Thus, $\leq$, its reflexive and transitive closure, is MSO-definable.

To show that the colouring is also definable we prove that for each colour c of χ there is a formula $\varphi_c(x, y)$ such that

$$(\mathfrak{A}, \chi) \models \varphi_c(x, y) \quad \text{iff} \quad x \leq y \text{ and } \chi(x, y) = c.$$

If $x \vdash y$ then there is a relation R and a tuple $\bar{a} \in R$ with $x, y \in \bar{a}$. Hence, $\chi(x, y)$ is available in $(\mathfrak{A}, \chi)$. Thus, there is a formula $\varphi_c^\vdash(x, y)$ which expresses that $x \vdash y$ and $\chi(x, y) = c$. We have $x \leq y$ iff there is a path $x = z_0 \vdash \cdots \vdash z_n = y$. Note that $z_i \vdash z_{i+1}$ implies $z_i/I \preceq z_{i+1}/I$. Therefore, we can compute $\chi(x, z_{i+1})$ from $\chi(x, z_i)$ and $\chi(z_i, z_{i+1})$. □

The proof of the converse is more involved and requires an investigation of the branching structure of a tuple.

Definition 21. Let $\bar{a}, \bar{b} \in A^n$. We say that $\bar{a}$ is a *reduct* of $\bar{b}$ iff

(1) the branching structures of $\bar{a}$ and $\bar{b}$ are the same,
(2) $\inf_\preceq \bar{a} \sim \inf_\preceq \bar{b}$,
(3) $(a_i \sqcap a_j)^{-1}(a_k \sqcap a_l) \sim (b_i \sqcap b_j)^{-1}(b_k \sqcap b_l)$ for all indices such that $a_i \sqcap a_j \preceq a_k \sqcap a_l$,
(4) $|a_i| < |\inf_\preceq \bar{a}| + nI$ for all $i < n$.

A tuple is called *reduced* if it is a reduct of itself.

Lemma 15. *If $\bar{a}$ is a reduct of $\bar{b}$ and $\bar{b} \in R$ then $\bar{a} \in R$.*

To check whether a tuple $\bar{a}$ belongs to a relation R we use the characterisation of Proposition 7. To do so we need the $\sim$-class of $u^{-1}v$ for branching points u and v of $\bar{a}$.

Definition 22. Let $\bar{a} \in A^n$. The elements $b_{ik} \in A$, for $i, k < n$, *code the branching structure* of $\bar{a}$ if

(1) $b_{ii} = a_i$ for $i < n$,
(2) $b_{ik}/nI \prec a_i \sqcap a_k \preceq b_{ik}$ for all i, k, and
(3) if $a_i \sqcap a_k \prec a_i \sqcap a_l$ then $b_{ik} \vdash b_{il}$ for $i, k, l < n$.

If we are given b_{ik} and b_{il} we can compute the $\sim$-class of $(a_i \sqcap a_k)^{-1}(a_i \sqcap a_l)$. Hence, if we can show that such elements always exists and that they are definable, then we are almost done.

Lemma 16. *For each branching structure $\mathfrak{X}$ there is a formula $\beta_{\mathfrak{X}}(\bar{x}, \bar{y})$ such that $(A, \leq, \chi) \models \beta(\bar{a}, \bar{b})$ if and only if the branching structure of $\bar{a}$ is $\mathfrak{X}$ and it is coded by $\bar{b}$.*

Proof. For all $i, k < n$ we have to express that $b_{ik}/m = a_i \sqcap a_k$ for some $m < nI$. Since $b_{ik} \vdash b_{ii} = a_i$ and $b_{ik} \vdash b_{kk} = a_k$ this can be determined by looking at $\chi(b_{ik}, a_i)$ and $\chi(b_{ik}, a_k)$. The verification of the other conditions can be done easily. $\square$

Lemma 17. *Let R be an n-ary relation of $\mathfrak{A}$ and $\bar{a} \in R$. There are elements $b_{ik} \in A$, $i, k < n$, coding the branching structure of $\bar{a}$.*

Proof. W.l.o.g. assume that $\Sigma = \{0, 1\}$. Let

$$J_{ik} := \{\, j < n \mid a_i \sqcap a_k \prec a_i \sqcap a_j \,\}.$$

We define tuples $\bar{c}_{ik}$, for $i, k < n$, by induction on $|J_{ik}|$ such that $\bar{c}_{ik}|_{J_{ik}}$ is a reduct of $\bar{a}|_{J_{ik}}$. If $J_{ik} = \emptyset$ let $\bar{c}_{ik} := \bar{a}$. Otherwise, let j, l be indices such that the branching points $a_i \sqcap a_l$ and $a_j \sqcap a_k$ are the immediate successors of $a_i \sqcap a_k$. Let

$$\bar{d}_{ik} := \bar{c}_{il}|_{J_{ik}} \cup \bar{c}_{jk}|_{J_{ki}} \cup \bar{a}|_{\overline{J_{ik} \cup J_{ki}}}.$$

Choose $\bar{c}_{ik}$ such that

$$\bar{c}_{ik}|_{J_{ik} \cup J_{ki}} \text{ is a reduct of } \bar{d}_{ik}|_{J_{ik} \cup J_{ki}} \quad \text{and} \quad \bar{c}_{ik}|_{\overline{J_{ik} \cup J_{ki}}} = \bar{a}|_{\overline{J_{ik} \cup J_{ki}}}.$$

Finally, set $b_{ik} := (c_{ik})_k$. Since, by construction, $\bar{d}_{ik}|_{\overline{J_{ki}}} \cup \bar{c}|_{J_{ki}} \in R$ we have

$$b_{ik} = (c_{ik})_k \vdash (d_{ik})_l = (c_{il})_l = b_{il}. \qquad \square$$

At last, we are able to prove the other direction.

Lemma 18. *The structure $(\mathfrak{A}, \chi)$ is* MSO*-definable in $(A, \leq, \chi)$ for every $\chi \geq \chi_{\sim}^{nI}$ where n is the maximal arity of relations of $\mathfrak{A}$.*

Proof. Let R be an n-ary relation of $(\mathfrak{A}, \chi)$. We prove the claim by induction on n.

($n = 1$) $R \subseteq \Sigma^*$ is regular with a coarser congruence than $\sim$. Thus, we can determine whether $a \in R$ by looking at $\chi(a)$.

($n > 1$) W.l.o.g. assume that all tuples $\bar{a} \in R$ have the same branching structure. For all branching points $a_i \sqcap a_k$ with immediate successor $a_i \sqcap a_l$ let W_{ikl} the the regular language such that $\bar{a} \in R$ iff $(a_i \sqcap a_k)^{-1}(a_i \sqcap a_l) \in W_{ikl}$ for all such i, k, l. By the preceding lemma it follows that $\bar{a} \in R$ if and only if there are elements b_{ik}, $i, k < n$, coding the branching structure of $\bar{a}$ and constants m_{ik} such that $b_{ik}/m_{ik} = a_i \sqcap a_k$ and $(b_{ik}/m_{ik})^{-1}(b_{il}/m_{il}) \in W_{ikl}$ for all admissible i, k, l. Since $b_{ik} \vdash b_{il}$ we can check the latter condition by looking at $\chi(b_{ik}, b_{il})$. □

Theorem 23. *Let $\mathfrak{A}$ be a tree-interpretable structure. There is a congruence colouring χ_0 such that $(\mathfrak{A}, \chi)$ is finitely* MSO($\exists^\kappa$)*-axiomatisable for all $\chi \geq \chi_0$.*

Proof. The proof is completely analogous to the one of Theorem 19. Let χ_0' be the colouring of Theorem 19 for the structure $(A, \leq)$, and set $\chi_0 := \chi_0' \sqcup \chi_\sim^{nI}$ where n is the maximal arity of relations of $\mathfrak{A}$. Let $\mathfrak{X}$ be a structure. By the preceding lemmas $(\mathfrak{A}, \chi)$ and $(A, \leq, \chi)$ are MSO-definable within each other. Let $\mathcal{I} : (A, \leq, \chi) \leq_{\text{MSO}} (\mathfrak{A}, \chi)$ be the corresponding interpretation. It follows that $\mathfrak{X} \cong (\mathfrak{A}, \chi)$ iff $\mathcal{I}(\mathfrak{X}) \cong \mathcal{I}(\mathfrak{A}, \chi)$. The later condition is MSO($\exists^\kappa$)-expressible by Theorem 19. □

Since GSO($\exists^\kappa$) allows quantification over colourings χ we obtain as immediate corollary the following result.

Theorem 24. *Every tree-interpretable structure is finitely* GSO($\exists^\kappa$)*-axiomatisable.*

6.5 Lower bounds

We have shown that every tree-interpretable structure is finitely GSO($\exists^\kappa$)-axiomatisable. Of course, the question arises if we can do better. In this section we show that at least the quantifiers $\exists^{\aleph_0}$ and $\exists^{\aleph_1}$ are needed. Since all tree-interpretable structures are countable we obviously can do without the ones for higher cardinalities.

For a logic $\mathfrak{L}$ let $\mathfrak{L}_m$ denote the set of $\mathfrak{L}$-formulae of quantifier rank at most m where we count both first- and second-order quantifiers. The following statements about the expressivity of MSO$_m$ and MSO$_m(\exists^{\aleph_0})$ can easily be proved using the corresponding versions of the Ehrenfeucht-Fraïssé game.

Lemma 19. (a) *For every $m \in \mathbb{N}$ there exists a constant k such that two sets A and B are* MSO$_m$*-equivalent if and only if either*

$$|A| = |B| \quad \textit{or} \quad |A|, |B| \geq k.$$

(b) *For every $m \in \mathbb{N}$ there exists a constant k such that two sets A and B are* MSO$_m(\exists^{\aleph_0})$*-equivalent if and only if either*

$$|A| = |B|, \quad \textit{or} \quad k \leq |A|, |B| < \aleph_0, \quad \textit{or} \quad |A|, |B| \geq \aleph_0.$$

(c) *Any two infinite sets are* MSO($\exists^{\aleph_0}$)*-equivalent.*

Lemma 20. *For all* GSO($\exists^\kappa$)*-sentences* φ *there is an* MSO($\exists^\kappa$)*-sentence* φ' *such that*

$$\mathfrak{T} \models \varphi \quad \text{iff} \quad \mathfrak{T} \models \varphi' \qquad \textit{for every tree } \mathfrak{T}.$$

Proof. Since each vertex has at most one predecessor one can code a set of edges by the set of their second components. This way each quantifier over sets of edges can be replaced by a monadic quantifier. □

Theorem 25. *There exists tree-interpretable trees which are not* GSO($\exists^{\aleph_0}$)*-axiomatisable.*

Proof. The preceding lemmas imply that $K_{1,\aleph_0} \equiv_{\mathrm{GSO}(\exists^{\aleph_0})} K_{1,\aleph_1}$. But the former structure is tree-interpretable while the latter obviously is not. □

This shows that we cannot do without all cardinality quantifiers even if we allow infinitely many axioms. But do we really need non-monadic second-order quantifiers?

Open Problem. Are there tree-interpretable structures which are not (finitely) MSO($\exists^\kappa$)-axiomatisable?

7 Automorphisms of tree-interpretable structures

As mentioned in the introduction the axiomatisation of a tree-interpretable structure can be used to investigate its automorphism group.

Lemma 21. *Let* $\mathfrak{A}$ *be a tree-interpretable structure and* $a \in A$*. The orbit* O *of* a *under automorphisms is* GSO($\exists^\kappa$)*-definable.*

Proof. If $\mathfrak{A}$ is tree-interpretable then so is $(\mathfrak{A}, a)$. Let $\varphi(x)$ be the GSO($\exists^\kappa$)-formula obtained from the axiom of $(\mathfrak{A}, a)$ by replacing every occurrence of the constant a by the variable x. It follows that

$$b \in O \quad \text{iff} \quad (\mathfrak{A}, b) \cong (\mathfrak{A}, a) \quad \text{iff} \quad \mathfrak{A} \models \varphi(b). \qquad \square$$

Lemma 22 (Pélecq [22])**.** *Let* $\mathfrak{A}$ *be a tree-interpretable structure of finite tree-width and let* O *be the orbit of* $a \in A$ *under automorphisms. Then* $(\mathfrak{A}, O)$ *is tree-interpretable.*

Proof. O is GSO($\exists^\kappa$)-definable by the preceding lemma. Since $\mathfrak{A}$ is of finite tree-width it follows that O is even MSO($\exists^\kappa$)-definable and, therefore, $(\mathfrak{A}, O) \leq_{\mathrm{MSO}} \mathfrak{T}_2$. □

We conclude this article with a simple application to the isomorphism problem.

Theorem 26 (Courcelle [8])**.** *Given two tree-interpretable structures* $\mathfrak{A}$ *and* $\mathfrak{B}$ *of finite-tree width one can decide whether* $\mathfrak{A} \cong \mathfrak{B}$*.*

Proof. Although not explicitly stated, the construction of the axiom in the previous section is effective. Thus, in order to determine whether $\mathfrak{A} \cong \mathfrak{B}$ one can construct the GSO($\exists^\kappa$)-formula $\varphi_{\mathfrak{A}}$ which axiomatises $\mathfrak{A}$ and check whether $\mathfrak{B}$ satisfies $\varphi_{\mathfrak{A}}$. □

Open Problem. Is isomorphism decidable for all tree-interpretable structures?

References

1. K. Barthelmann, *On equational simple graphs*, Tech. Rep. 9, Universität Mainz, Institut für Informatik, 1997.
2. ———, *When can an equational simple graph be generated by hyperedge replacement?*, LNCS, 1450 (1998), pp. 543–552.
3. A. Blumensath, *Automatic Structures*, Diploma Thesis, RWTH Aachen, 1999.
4. A. Blumensath and E. Grädel, *Automatic structures*, in Proc. 15th IEEE Symp. on Logic in Computer Science, 2000, pp. 51–62.
5. O. Burkart, *Model checking rationally restricted right closures of recognizable graphs*, ENTCS, 9 (1997).
6. D. Caucal, *On infinite transition graphs having a decidable monadic theory*, LNCS, 1099 (1996), pp. 194–205.
7. B. Courcelle, *The monadic second-order logic of graphs II: Infinite graphs of bounded width*, Math. System Theory, 21 (1989), pp. 187–221.
8. ———, *The monadic second-order logic of graphs IV: Definability properties of equational graphs*, Annals of Pure and Applied Logic, 49 (1990), pp. 193–255.
9. ———, *The monadic second-order logic of graphs VII: Graphs as relational structures*, Theoretical Computer Science, 101 (1992), pp. 3–33.
10. ———, *The monadic second-order logic of graphs VI: On several representations of graphs by relational structures*, Discrete Applied Mathematics, 54 (1994), pp. 117–149.
11. ———, *Structural properties of context-free sets of graphs generated by vertex replacement*, Information and Computation, 116 (1995), pp. 275–293.
12. ———, *Clique-width of countable graphs: A compactness property.* unpublished, 2000.
13. ———, *The monadic second-order logic of graphs XIV: Uniformly sparse graphs and edge set quantifications*, Theoretical Computer Science, (to appear).
14. Y. L. Ershov, S. S. Goncharov, A. Nerode, and J. B. Remmel, *Handbook of Recursive Mathematics*, North-Holland, 1998.
15. E. Grädel, C. Hirsch, and M. Otto, *Back and forth between guarded and modal logics*, in Proc. 15th IEEE Symp. on Logic in Computer Science, 2000, pp. 217–228.
16. B. Khoussainov and A. Nerode, *Automatic presentations of structures*, LNCS, 960 (1995), pp. 367–392.
17. G. Kuper, L. Libkin, and J. Paredaens, *Constraint Databases*, Springer-Verlag, 2000.
18. O. Kupferman and M. Y. Vardi, *An automata-theoretic approach to reasoning about infinite-state systems*, LNCS, 1855 (2000), pp. 36–52.
19. C. Morvan, *On rational graphs*, LNCS, 1784 (1996), pp. 252–266.
20. D. E. Muller and P. E. Schupp, *Groups, the theory of ends, and context-free languages*, J. of Computer and System Science, 26 (1983), pp. 295–310.
21. ———, *The theory of ends, pushdown automata, and second-order logic*, Theoretical Computer Science, 37 (1985), pp. 51–75.
22. L. Pélecq, *Isomorphismes et automorphismes des graphes context-free, équationnels et automatiques*, Ph. D. Thesis, Université Bordeaux I, 1997.
23. D. Seese, *The structure of the models of decidable monadic theories of graphs*, Annals of Pure and Applied Logic, 53 (1991), pp. 169–195.
24. G. Sénizergues, *Decidability of bisimulation equivalence for equational graphs of finite out-degree*, in Proc. 39th Annual Symp. on Foundations of Computer Science, 1998, pp. 120–129.
25. C. Stirling, *Decidability of bisimulation equivalence for pushdown processes.* unpublished, 2000.
26. W. Thomas, *Languages, automata, and logic*, in Handbook of Formal Languages, G. Rozenberg and A. Salomaa, eds., vol. 3, Springer, New York, 1997, pp. 389–455.

Aachener Informatik-Berichte

This is a list of recent technical reports. To obtain copies of technical reports please consult http://aib.informatik.rwth-aachen.de/ or send your request to: Informatik-Bibliothek, RWTH Aachen, Ahornstr. 55, 52056 Aachen, Email: `biblio@informatik.rwth-aachen.de`

95-11 *	M. Staudt / K. von Thadden: Subsumption Checking in Knowledge Bases
95-12 *	G.V. Zemanek / H.W. Nissen / H. Hubert / M. Jarke: Requirements Analysis from Multiple Perspectives: Experiences with Conceptual Modeling Technology
95-13 *	M. Staudt / M. Jarke: Incremental Maintenance of Externally Materialized Views
95-14 *	P. Peters / P. Szczurko / M. Jeusfeld: Business Process Oriented Information Management: Conceptual Models at Work
95-15 *	S. Rams / M. Jarke: Proceedings of the Fifth Annual Workshop on Information Technologies & Systems
95-16 *	W. Hans / St. Winkler / F. Sáenz: Distributed Execution in Functional Logic Programming
96-1 *	Jahresbericht 1995
96-2	M. Hanus / Chr. Prehofer: Higher-Order Narrowing with Definitional Trees
96-3 *	W. Scheufele / G. Moerkotte: Optimal Ordering of Selections and Joins in Acyclic Queries with Expensive Predicates
96-4	K. Pohl: PRO-ART: Enabling Requirements Pre-Traceability
96-5	K. Pohl: Requirements Engineering: An Overview
96-6 *	M. Jarke / W. Marquardt: Design and Evaluation of Computer–Aided Process Modelling Tools
96-7	O. Chitil: The ς-Semantics: A Comprehensive Semantics for Functional Programs
96-8 *	S. Sripada: On Entropy and the Limitations of the Second Law of Thermodynamics
96-9	M. Hanus (Ed.): Proceedings of the Poster Session of ALP'96 — Fifth International Conference on Algebraic and Logic Programming
96-10	R. Conradi / B. Westfechtel: Version Models for Software Configuration Management
96-11 *	C. Weise / D. Lenzkes: A Fast Decision Algorithm for Timed Refinement
96-12 *	R. Dömges / K. Pohl / M. Jarke / B. Lohmann / W. Marquardt: PRO-ART/CE* — An Environment for Managing the Evolution of Chemical Process Simulation Models
96-13 *	K. Pohl / R. Klamma / K. Weidenhaupt / R. Dömges / P. Haumer / M. Jarke: A Framework for Process-Integrated Tools

96-14 * R. Gallersdörfer / K. Klabunde / A. Stolz / M. Eßmajor: INDIA — Intelligent Networks as a Data Intensive Application, Final Project Report, June 1996

96-15 * H. Schimpe / M. Staudt: VAREX: An Environment for Validating and Refining Rule Bases

96-16 * M. Jarke / M. Gebhardt, S. Jacobs, H. Nissen: Conflict Analysis Across Heterogeneous Viewpoints: Formalization and Visualization

96-17 M. Jeusfeld / T. X. Bui: Decision Support Components on the Internet

96-18 M. Jeusfeld / M. Papazoglou: Information Brokering: Design, Search and Transformation

96-19 * P. Peters / M. Jarke: Simulating the impact of information flows in networked organizations

96-20 M. Jarke / P. Peters / M. Jeusfeld: Model-driven planning and design of cooperative information systems

96-21 * G. de Michelis / E. Dubois / M. Jarke / F. Matthes / J. Mylopoulos / K. Pohl / J. Schmidt / C. Woo / E. Yu: Cooperative information systems: a manifesto

96-22 * S. Jacobs / M. Gebhardt, S. Kethers, W. Rzasa: Filling HTML forms simultaneously: CoWeb architecture and functionality

96-23 * M. Gebhardt / S. Jacobs: Conflict Management in Design

97-01 Jahresbericht 1996

97-02 J. Faassen: Using full parallel Boltzmann Machines for Optimization

97-03 A. Winter / A. Schürr: Modules and Updatable Graph Views for PROgrammed Graph REwriting Systems

97-04 M. Mohnen / S. Tobies: Implementing Context Patterns in the Glasgow Haskell Compiler

97-05 * S. Gruner: Schemakorrespondenzaxiome unterstützen die paargrammatische Spezifikation inkrementeller Integrationswerkzeuge

97-06 M. Nicola / M. Jarke: Design and Evaluation of Wireless Health Care Information Systems in Developing Countries

97-07 P. Hofstedt: Taskparallele Skelette für irregulär strukturierte Probleme in deklarativen Sprachen

97-08 D. Blostein / A. Schürr: Computing with Graphs and Graph Rewriting

97-09 C.-A. Krapp / B. Westfechtel: Feedback Handling in Dynamic Task Nets

97-10 M. Nicola / M. Jarke: Integrating Replication and Communication in Performance Models of Distributed Databases

97-13 M. Mohnen: Optimising the Memory Management of Higher-Order Functional Programs

97-14 R. Baumann: Client/Server Distribution in a Structure-Oriented Database Management System

97-15 G. H. Botorog: High-Level Parallel Programming and the Efficient Implementation of Numerical Algorithms

98-01 * Jahresbericht 1997

98-02 S. Gruner/ M. Nagel / A. Schürr: Fine-grained and Structure-oriented Integration Tools are Needed for Product Development Processes

98-03 S. Gruner: Einige Anmerkungen zur graphgrammatischen Spezifikation von Integrationswerkzeugen nach Westfechtel, Janning, Lefering und Schürr

98-04 * O. Kubitz: Mobile Robots in Dynamic Environments

98-05 M. Leucker / St. Tobies: Truth — A Verification Platform for Distributed Systems

98-07 M. Arnold / M. Erdmann / M. Glinz / P. Haumer / R. Knoll / B. Paech / K. Pohl / J. Ryser / R. Studer / K. Weidenhaupt: Survey on the Scenario Use in Twelve Selected Industrial Projects

98-08 * H. Aust: Sprachverstehen und Dialogmodellierung in natürlichsprachlichen Informationssystemen

98-09 * Th. Lehmann: Geometrische Ausrichtung medizinischer Bilder am Beispiel intraoraler Radiographien

98-10 * M. Nicola / M. Jarke: Performance Modeling of Distributed and Replicated Databases

98-11 * A. Schleicher / B. Westfechtel / D. Jäger: Modeling Dynamic Software Processes in UML

98-12 * W. Appelt / M. Jarke: Interoperable Tools for Cooperation Support using the World Wide Web

98-13 K. Indermark: Semantik rekursiver Funktionsdefinitionen mit Striktheitsinformation

99-01 * Jahresbericht 1998

99-02 * F. Huch: Verifcation of Erlang Programs using Abstract Interpretation and Model Checking — Extended Version

99-03 * R. Gallersdörfer / M. Jarke / M. Nicola: The ADR Replication Manager

99-04 M. Alpuente / M. Hanus / S. Lucas / G. Vidal: Specialization of Functional Logic Programs Based on Needed Narrowing

99-07 Th. Wilke: CTL+ is exponentially more succinct than CTL

99-08 O. Matz: Dot-Depth and Monadic Quantifier Alternation over Pictures

2000-01 * Jahresbericht 1999

2000-02 Jens Vöge / Marcin Jurdziński: A Discrete Strategy Improvement Algorithm for Solving Parity Games

2000-04 Andreas Becks, Stefan Sklorz, Matthias Jarke: Exploring the Semantic Structure of Technical Document Collections: A Cooperative Systems Approach

2000-05 * Mareike Schoop: Cooperative Document Management

2000-06 * Mareike Schoop, Christoph Quix (Ed.): Proceedings of the Fifth International Workshop on the Language-Action Perspective on Communication Modelling

2000-07 * Markus Mohnen / Pieter Koopman (Eds.): Proceedings of the 12th International Workshop of Functional Languages

2000-08 Thomas Arts / Thomas Noll: Verifying Generic Erlang Client-Server Implementations

2001-01 * Jahresbericht 2000

2001-02 Benedikt Bollig / Martin Leucker: Deciding LTL over Mazurkiewicz Traces

2001-03 Thierry Cachat: The power of one-letter rational languages

2001-04 Benedikt Bollig / Martin Leucker / Michael Weber: Local Parallel Model Checking for the Alternation free μ-calculus

2001-05 Benedikt Bollig / Martin Leucker / Thomas Noll: Regular MSC languages

2001-06 Achim Blumensath: Prefix-Recognisable Graphs and Monadic Second-Order Logic

2001-07 Martin Grohe / Stefan Wöhrle: An Existential Locality Theorem

2001-08 Mareike Schoop / James Taylor (eds.): Proceedings of the Sixth International Workshop on the Language-Action Perspective on Communication Modelling

2001-09 Thomas Arts / Jürgen Giesl: A collection of examples for termination of term rewriting using dependency pairs

* These reports are only available as a printed version.

Please contact `biblio@informatik.rwth-aachen.de` to obtain copies.