Trees over Infinite Structures and Path Logics with Synchronization

Alex Spelten Wolfgang Thomas Sarah Winter
RWTH Aachen University
Germany
{spelten,thomas,winter}@automata.rwth-aachen.de

We provide decidability and undecidability results on the model-checking problem for infinite tree structures. These tree structures are built from sequences of elements of infinite relational structures. More precisely, we deal with the tree iteration of a relational structure $\mathscr{M}$ in the sense of Shelah-Stupp. In contrast to classical results, where model-checking is shown decidable for MSO-logic, we show decidability of the tree model-checking problem for logics that allow only path quantifiers and chain quantifiers (where chains are subsets of paths), as they appear in branching time logics; however, at the same time, the tree is enriched by the equal-level relation (which holds between vertices u, v if they are on the same tree level). We separate cleanly the tree logic from the logic used for expressing properties of the underlying structure $\mathscr{M}$. We illustrate the scope of the decidability results by showing that two slight extensions of the framework lead to undecidability. In particular, this applies to the (stronger) tree iteration in the sense of Muchnik-Walukiewicz.

1 Introduction

A key result in the field of "infinite-state model-checking" is Rabin's Tree Theorem [10]. It says that the monadic second-order theory (short: MSO-theory) of the binary tree is decidable. Many decidability results on theories of infinite structures have been obtained by a reduction to Rabin's Tree Theorem. It is also well-known that a slight extension of the signature of the binary tree leads to undecidability: The expansion of the binary tree by the "equal-level relation" E has an undecidable monadic theory.

The situation changes when set quantification is restricted to "chains", i.e., sets that are linearly ordered by the partial tree ordering. It is known ([16]) that for the unlabeled binary tree and also for any regular binary tree, the chain logic theory of the tree is decidable in the presence of E. This result is of interest in verification since a large number of logical concepts that occur in specifications of nonterminating systems refer to computation paths and their subsets (i.e., to chains), for example in branching time logics. The second-order quantifiers in these applications do not refer to global colorings of computation trees (for which monadic logic would be invoked) but rather to quantification over chains. The equal-level relation adds the feature of synchronization to computation paths.

In recent years, a theory of words and trees over infinite alphabets emerged ([8, 2, 4]) that opens a way for generalizations. Here, a computation path is a sequence of letters chosen from a relational structure $\mathscr{M} = (M, R_1, \dots, R_k)$, which is infinite in general, rather than from a finite alphabet Σ. Instead of the binary tree obtained from the words of $\{0,1\}^*$ built from the two element alphabet $\{0,1\}$, the infinitely branching infinite tree with vertices in M^* is considered.

There are two fundamental constructions of a tree structure built from an "alphabet structure" $\mathscr{M}$, called "weak", respectively "strong" tree iteration of $\mathscr{M}$, and denoted here $\mathscr{M}^\#$, respectively $\mathscr{M}^*$. For $\mathscr{M} = (M, R_1, \dots, R_k)$, let

$$\mathscr{M}^\# = (M^*, S, \preceq, R_1^*, \dots, R_k^*)$$

To appear in EPTCS.

where $S(u,v)$ holds if $v = um$ for some $u \in M^*, m \in M$, $\preceq$ is the reflexive transitive closure of S, and, for ℓ-ary R_i, we have $R_i^*(v_1,\dots,v_\ell)$ iff for some $z \in M^*$, $v_j = zm_j$ for $j = 1,\dots,\ell$ such that $R_i(m_1,\dots,m_\ell)$ holds in $\mathscr{M}$. This iteration is also called Shelah-Stupp iteration, going back to [13, 14].

The strong tree iteration $\mathscr{M}^*$ is obtained from the weak one by adjoining the "clone predicate"

$$C = \{u\ m\ m \mid u \in M^*, m \in M\}$$

to the signature. It allows to connect two levels of the tree structure in a way that "unfolding" becomes definable.

As shown by Shelah and Stupp [13, 14], respectively Muchnik and Walukiewicz (see the announcement in [12] and the proof in [19]), the MSO-theory of $\mathscr{M}^\#$ and the MSO-theory of $\mathscr{M}^*$ are decidable if the MSO-theory of $\mathscr{M}$ is. In the present paper we show the decidability of the chain logic theory of structures $\mathscr{M}_E^\#$, obtained by adjoining the equal level relation E to $\mathscr{M}^\#$, under mild assumptions on the structure $\mathscr{M}$. Our results extend work of Kuske and Lohrey [6] on structures $\mathscr{M}^\#$ and of Bès [1] on structures $\mathscr{M}_E^\#$. Furthermore, we show – in contrast to the Muchnik-Walukiewicz result for MSO-logic – that a transfer of this decidability result to tree structures $\mathscr{M}_E^*$ is not possible.

Bès shows the decidability of the chain logic theory of $\mathscr{M}_E^\#$ if the first-order theory of $\mathscr{M}$ is decidable. Here we refine his result: We refer to any logic $\mathscr{L}$ such that the $\mathscr{L}$-theory of $\mathscr{M}$ is decidable, and we consider an *extension* of the chain theory of $\mathscr{M}^\#$ in which further quantifications are allowed, namely quantifiers of $\mathscr{L}$ restricted to the set of siblings of any element z. (Thus one allows quantifiers over elements y that are S-successors of any given element z.) We call the corresponding theory the *chain logic theory of $\mathscr{M}_E^\#$ with $\mathscr{L}$ on siblings*. We show that this theory is decidable if the $\mathscr{L}$-theory of $\mathscr{M}$ is.

In our framework two logics play together: The logic $\mathscr{L}$ allows to express relations between $\mathscr{M}$-elements as they appear as sons of some given node of the tree, and chain logic is used to speak about (sets of) tree elements arranged along paths. Referring to the standard graphical representation of trees, $\mathscr{L}$ captures the horizontal dimension and chain logic the vertical dimension. On the level of signatures, the predicate E of the tree signature refers to the horizontal while the successor and the prefix relation refer to the vertical aspect; finally, the signature of $\mathscr{M}$ enters in the horizontal dimension, restricted to the children of a tree node.

Standard examples of logics $\mathscr{L}$ are first-order logic FO, monadic second-order logic MSO and its weak fragment WMSO, transitive closure logic TC, or extensions of FO by counting operators. (In this paper we do not present a precise definition of the concept of a "logic" and just refer the reader to [5].) Standard examples of models $\mathscr{M}$ originate in arithmetic and analysis, e.g. $(\mathbb{N},+,<,0), (\mathbb{R},+,<,0,1), (\mathbb{R},+,\cdot,<,0,1)$ (whose first-order theory is decidable). In applications, one may work with structures $\mathscr{M}$ that are direct products of finite transition graphs with infinite value structures such as $(\mathbb{R},+,<,0,1)$ or the real field $(\mathbb{R},+,\cdot,<,0,1)$.

The method to show the main result rests on a simple observation, first exploited in [16]: Consider the tree with domain M^* where M is ordered of order type ω. A formula $\varphi(X_1\dots,X_n)$ of chain logic – with chains c_i as possible interpretations of the X_i – can be viewed as a statement about $2n$-tuples of ω-words as follows. Any single chain c_i is encoded by two ω-words; the first is from M^ω and describes the (leftmost) full path of which c_i is a subset. The second is a 0-1-sequence describing by its entries 0 and 1 which elements of the path belong to c_i. Now the obtained $2n$-tuple of ω-words over M, respectively $\{0,1\}$, can be viewed as a single ω-word with alphabet letters from $(M\times\{0,1\})^n$. Using this translation of n-tuples of chains of $\mathscr{M}^*$ into ω-words over $(M\times\{0,1\})^n$, we obtain a translation of chain logic formulas into MSO-formulas interpreted in ω-words, i.e., structures with domain $\mathbb{N}$. More precisely, when $\mathscr{L}$ is the logic used for $\mathscr{M}$, we obtain a formula of "$\mathscr{M}$-$\mathscr{L}$-MSO".

This framework of $\mathscr{M}$-$\mathscr{L}$-MSO is in turn equivalent to Büchi automata (over ω-words with entries from $(M \times \{0,1\})^n$). We develop these $\mathscr{M}$-$\mathscr{L}$-Büchi automata as a preparation for the main result. It turns out that these automata allow closure and decidability results in precise analogy to the classical theory over finite alphabets. As a consequence we obtain that the chain theory of $\mathscr{M}^{\#}_E$ with $\mathscr{L}$ on siblings is decidable if the $\mathscr{L}$-theory of $\mathscr{M}$ is.

While the setting of $\mathscr{M}$-$\mathscr{L}$-Büchi automata is sufficient for the study of tree models $\mathscr{M}^{\#}_E$, it has to be extended to cope with strong tree iterations $\mathscr{M}^{*}_E$ where the clone predicate enters. We define "strong $\mathscr{M}$-$\mathscr{L}$-Büchi automata" for this purpose. Here a remarkable difference occurs between the cases of an input alphabet M (with infinite M) and an input alphabet M^n for $n > 1$. We give a brief explanation that in the first case strong Büchi automata behave as $\mathscr{M}$-$\mathscr{L}$-Büchi automata (however using just $\mathscr{L} = \text{MSO}$), whereas in the second case of input alphabets M^n with $n > 1$, undecidability phenomena enter (in the form that the emptiness problem becomes undecidable). Along this line we show that the chain theory (and even the first-order theory) of $\mathscr{M}^{*}_E$ is undecidable if $\mathscr{M}$ is infinite – in fact already for the case that $\mathscr{M}$ is the successor structure of the natural numbers.

A last result of the paper shows that the decidability result (on the chain theory of $\mathscr{M}^{\#}_E$ with $\mathscr{L}$ on siblings) also fails when quantification extends over an entire tree level rather than just siblings of a fixed node. We obtain this for the weak tree iteration of the two element alphabet $\{0,1\}$ when the logic $\mathscr{L}$ is MSO.

The paper is structured as follows. In the subsequent section we collect the necessary terminology. Section 3 develops the theory of Büchi automata over ω-words whose letters are n-tuples from an infinite structure $\mathscr{M}$ and using a logic $\mathscr{L}$ to specify properties of such letters in $\mathscr{M}$. In Section 4 we deduce the decidability of the chain theory of $\mathscr{M}^{\#}_E$ with $\mathscr{L}$ on siblings when the $\mathscr{L}$-theory of $\mathscr{M}$ is decidable. Section 5 gives the two mentioned undecidability results. We conclude with remarks on further work.

2 Terminology

We consider relational structures with finite signature. Such a structure is presented in the format $\mathscr{M} = (M, R_1, \ldots, R_k)$ where R_i is of arity $r_i > 0$. We focus on structures called "admissible": In this case there are two designated elements (usually called 0 and 1), represented by two singleton predicates P_0, P_1 that belong to the tuple $(R_1, \ldots, R_k)$. Then we can view bit sequences as special sequences over $\mathscr{M}$.

For an ω-word $\alpha \in \Sigma^\omega$ (where Σ may be infinite), written $\alpha = \alpha(0)\alpha(1)\ldots$, we denote by $\alpha[i, j]$ the segment $\alpha(i)\ldots\alpha(j)$.

We introduce two tree models built from a relational structure $\mathscr{M}$. The first is the weak tree iteration

$$\mathscr{M}^{\#} = (M^*, \preceq, S, R^*_1, \ldots, R^*_k)$$

where $u \preceq v :\Leftrightarrow u$ is a prefix of v, S is the successor relation containing all pairs (u, um) with $u \in M^*, m \in M$, and for every R_i, say of arity ℓ, we have $R^*_i(v_1, \ldots, v_\ell)$ iff there exists $z \in M^*$, $m_1, \ldots m_\ell \in M$ such that $v_j = zm_j$ for $j = 1, \ldots m_\ell$ and $R_i(m_1, \ldots, m_\ell)$. (In [1] a variant of this definition is used, namely that there exist $z_1, \ldots, z_\ell \in M^*$ of same length and $m_1, \ldots, m_\ell \in M$ such $v_j = z_j m_j$ with $R_i(m_1, \ldots, m_\ell)$.)

As mentioned in the introduction, the strong tree iteration of $\mathscr{M}$ is the structure

$$\mathscr{M}^* = (M^*, \preceq, S, R^*_1, \ldots, R^*_k, C)$$

where everything is as above for $\mathscr{M}^{\#}$ and $C = \{u\ m\ m \mid u \in M^*, m \in M\}$. The expansions of $\mathscr{M}^{\#}$, $\mathscr{M}^*$ by the equal level relation E (with $E(u, v)$ iff $|u| = |v|$) are denoted $\mathscr{M}^{\#}_E$, $\mathscr{M}^{*}_E$, respectively.

If $\mathscr{M}$ is finite, we assume that each individual letter of M is definable. The usual approach is to introduce a constant in the signature of $\mathscr{M}$ for each element of M. In the present paper we stick to relational structures and use a singleton predicate R_m for each element $m \in M$. So the binary alphabet $\{0,1\}$ is coded by the structure $\mathscr{M}_2 = (\{0,1\}, R_0, R_1)$ with $R_0 = \{0\}$, $R_1 = \{1\}$. In the case of finite structures $\mathscr{M}$ there is no essential difference between $\mathscr{M}^\#$ and $\mathscr{M}^*$, since the clone predicate C becomes definable in $\mathscr{M}^\#$ by the equivalence

$$C(v) \leftrightarrow \bigvee_{m \in M} (\exists u (R^*_m(u) \wedge S(u,v) \wedge R^*_m(v)).$$

Let us introduce chain logic over the tree structures $\mathscr{M}^\#$ and $\mathscr{M}^*$ built from $\mathscr{M}$. A path (through the tree domain M^*) is a maximal set linearly ordered by $\preceq$; it may be identified with an ω-word in M^ω, obtained as the common extension of all the words $u \in M^*$ forming the path. A chain is a subset of a path. So a singleton set in M^* is a chain, and we can easily simulate first-order quantification by quantification over chains restricted to singletons. We call chain logic the fragment of MSO logic in which set quantification is restricted to chains.

Sometimes it is convenient to eliminate first-order variables and quantifiers in terms of (singleton) chain quantifiers. This simplifies the setting since only one kind $X_1, X_2, \ldots$ of variables remains, ranging over chains. In order to simulate first-order logic, the signature of tree models has to be adapted. As atomic formulas one uses

- $\mathrm{Sing}(X)$ for "X is a singleton"
- $X_i \subseteq X_j$ with its standard meaning,
- $\mathrm{Succ}(X_i, X_j)$ for "X_i is a singleton $\{x_i\}$, X_j is a singleton $\{x_j\}$, with $S(x_i, x_j)$; similarly for $X_i \preceq X_j$.

The resulting formalism is called chain$_0$ logic; it has the same expressive power as chain logic.

For an admissible alphabet M (containing two identifiable elements 0,1) we encode a chain c as a pair $\hat{c} := (\alpha, \beta) \in (M^\omega)^2$ where

- α encodes the path of which c is a subset. As c can be finite, we set α to be the path $m_0 \ldots m_r 000 \ldots$ where m_r is the last c-element of which c is a subset; it can be interpreted as a sequence of "directions". Note that for each element w in c it holds that w is a prefix of α.
- β codes membership in c along the path α, i.e., $\beta(i) = 1$ iff $\alpha[0,i] \in c$.

So if $c = \emptyset$, α is the path 0^ω through the tree M^* and β also is the sequence that is constant 0.

The technical treatment below is simplified when viewing an n-tuple $(\alpha_1, \ldots, \alpha_n)$ of ω-words over M as a single ω-word over M^n, the *convolution* of $(\alpha_1, \ldots, \alpha_n)$:

$$\langle \alpha_1, \ldots, \alpha_n \rangle := \begin{bmatrix} \alpha_1(0) \\ \vdots \\ \alpha_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1(1) \\ \vdots \\ \alpha_n(1) \end{bmatrix} \cdots \quad \in (M^n)^\omega$$

Similarly, we define the *convolution of a relation* $R \subseteq (M^\omega)^n$ *of* ω*-words* to be the ω-language

$$L_R := \{\langle \alpha_1, \ldots, \alpha_n \rangle \mid (\alpha_1, \ldots, \alpha_n) \in R\}.$$

So the n-tuples of M-elements just considered will be used as letters of ω-words and input letters of Büchi automata. Transitions of automata will be specified in a logic $\mathscr{L}$ by means of $\mathscr{L}$-formulas $\varphi(x_1, \ldots, x_n)$. Each of these formulas defines a unary predicate $\varphi^{\mathscr{M}}$ over M^n:

$$\varphi^{\mathscr{M}} = \{(m_1, \ldots, m_n) \in M^n \mid \mathscr{M} \models \varphi[m_1, \ldots, m_n]\}$$

In general we consider ω-models over M^n for a signature that is given by a finite set Φ of $\mathscr{L}$-formulas: Given a tuple $(\alpha_1,\ldots,\alpha_n)$ of words over an alphabet M and a finite set Φ of $\mathscr{L}$-formulas $\varphi_1,\ldots,\varphi_k$ with n free variables each, we define the structure

$$\underline{\langle\alpha_1,\ldots,\alpha_n\rangle} = (\mathbb{N},0,<,S,(P_\varphi)_{\varphi\in\Phi})$$

with the usual interpretations of $0,<,S$ (the latter for the successor relation), and the letter predicates $P_{\varphi_j} = \{i\in\mathbb{N} \mid (\alpha_1(i),\ldots,\alpha_n(i)) \in \varphi_j^{\mathscr{M}}\}$. Thus, P_φ collects all letter positions of $\langle\alpha_1,\ldots,\alpha_n\rangle$ which carry a letter from M^n that shares the property described by φ.

For these ω-models over $\mathscr{M}$, equipped with predicates P_φ defined in $\mathscr{L}$, we shall use a generalized form of MSO-logic, where – as usual in ω-language theory – the first-order quantifiers range over $\mathbb{N}$ and the monadic second-order quantifiers over sequences of letters (here from M). The system will be called $\mathscr{M}$-$\mathscr{L}$-MSO.

For an $\mathscr{M}$-$\mathscr{L}$-MSO-sentence ψ, where the predicates P_φ are introduced via $\mathscr{L}$-formulas $\varphi(x_1,\ldots,x_n)$ with n free variables, we set

$$L(\psi) = \{\langle\alpha_1,\ldots,\alpha_n\rangle \in (M^n)^\omega \mid \underline{\langle\alpha_1,\ldots,\alpha_n\rangle} \models \psi\}$$

as the ω-language defined by ψ. We say a relation $R\subseteq (M^\omega)^n$ is *$\mathscr{M}$-$\mathscr{L}$-MSO definable* if there is a $\mathscr{M}$-$\mathscr{L}$-MSO sentence ψ with $L_R = L(\psi)$.

Later on, it will be convenient to refer to the component entries of an ω-word $\langle\alpha_1,\ldots,\alpha_n\rangle$ in a more readable way than via an index $i\in\{1,\ldots,n\}$. So, when a sequence variable Y is used for the i-th component α_i, we shall write $Y(s)$ to indicate the element $\alpha_i(s)$ for $s\in\mathbb{N}$.

Analogous definitions can be given for the case of finite words over M^n.

3 $\mathscr{M}$-$\mathscr{L}$-Büchi Automata

In this section we introduce finite automata over words and ω-words whose letters are n-tuples from M which is the domain of a (in general infinite) relational structure $\mathscr{M}$. Transitions of the automata are defined in a logic $\mathscr{L}$. Mentioning both parameters (the structure $\mathscr{M}$ and the logic $\mathscr{L}$), we speak of $\mathscr{M}$-$\mathscr{L}$-automata and $\mathscr{M}$-$\mathscr{L}$-Büchi automata. In the first subsection we obtain, not surprisingly, an equivalence between $\mathscr{M}$-$\mathscr{L}$-automata and $\mathscr{M}$-$\mathscr{L}$-MSO. In the second subsection we add some remarks on an extended model ("strong Büchi automata") that allows to capture the clone predicate between successive letters.

3.1 The standard case

Let $\mathscr{M}$ be a structure with domain M. An *$\mathscr{M}$-$\mathscr{L}$-Büchi automaton* over n-tuples of M-elements is of the form

$$\mathscr{B} = (Q,M^n,q_0,\Delta,F)$$

with a finite set Q of states, the input alphabet M^n, the initial state $q_0\in Q$, the set $F\subseteq Q$ of accepting states and the finite transition relation $\Delta\subseteq Q\times\Phi_n\times Q$, where Φ_n is the set of $\mathscr{L}$-formulas with n free variables.

Let us define acceptance of ω-words. If $\alpha = \langle\alpha_1,\ldots,\alpha_n\rangle$ is an ω-word over M^n, a *run* of $\mathscr{B}$ on α is an infinite sequence of states $\rho=\rho(0)\rho(1)\ldots$ with $\rho(0)=q_0$ such that for every $i\geq 0$ there exists an $\mathscr{M}$-$\mathscr{L}$-formula $\varphi(x_1,\ldots,x_n)$ and a transition $(\rho(i),\varphi,\rho(i+1))$ satisfying

$$\mathscr{M}\models\varphi[\alpha_1(i),\ldots,\alpha_n(i)]$$

A run ρ of $\mathscr{B}$ on α is *successful* if there exist infinitely many i such that $\rho(i) \in F$. We say that $\mathscr{B}$ *accepts* α if there exists a successful run of $\mathscr{B}$ on α. We denote by $L(\mathscr{B})$ the set of ω-words over M^n accepted by $\mathscr{B}$.

Similarly, we define $\mathscr{M}$-$\mathscr{L}$-automata for the case of finite words (as done in [1]). Languages accepted by these automata will be denoted as $\mathscr{M}$-$\mathscr{L}$-recognizable languages. We note some basic properties.

Lemma 1

- *The class of $\mathscr{M}$-$\mathscr{L}$-recognizable languages (of finite words) is closed under union, projection, and complementation.*
- *For an $\mathscr{M}$-$\mathscr{L}$-recognizable language (of finite words) $U \subseteq (M^n)^*$ and an $\mathscr{M}$-$\mathscr{L}$-Büchi recognizable ω-language $K \subseteq (M^n)^\omega$, we have*
  1. *U^ω is $\mathscr{M}$-$\mathscr{L}$-Büchi recognizable.*
  2. *$U \cdot K$ is $\mathscr{M}$-$\mathscr{L}$-Büchi recognizable.*

Proof The closure properties of $\mathscr{M}$-$\mathscr{L}$-recognizable languages (of finite words) are shown by slight adaptions of the classical case (where the alphabet is finite). Here, we concentrate on pointing out the adaptions rather than the actual constructions. For example, an automaton for the projection from M^n to M^{n-1} can easily be obtained by replacing the “label” $\varphi(x_1,\dots,x_n)$ of a transition by $\exists x_n \varphi(x_1,\dots,x_n)$. For the complementation, we follow the strategy of a determinization via a powerset construction and then simply swapping the sets F and $Q \setminus F$ (as outlined in [1]). The idea is as follows: Given an $\mathscr{M}$-$\mathscr{L}$ automaton $\mathscr{B}$ (on finite words), $\mathscr{B}$ does not necessarily provide a run (accepting or not accepting) for every possible input letter in M^n, i.e., there might be a letter that does not satisfy any of the formulas of the transitions. For the construction of the complement automaton, one modifies the set of formulas for the transitions such that each input word leads to a complete run, and additionally, one prepares for determinism: Let $\varphi_1,\dots,\varphi_m$ be the formulas which occur in the transitions of $\mathscr{B}$. For each subset $J \subseteq \{1,\dots,m\}$, introduce the formula $\psi_J := \bigwedge_{i \in J} \varphi_i \wedge \bigwedge_{i \notin J} \neg\varphi_i$. Note that for $J \neq K$, there is no symbol $\overline{m} \in M^n$ with $\mathscr{M} \models \psi_J \wedge \psi_K[\overline{m}]$, and for each $\overline{m}$, there is a set J such that $\mathscr{M} \models \psi_J[\overline{m}]$. Then we construct $\mathscr{B}'$ by replacing each transition $(p,\varphi_i,q) \in \Delta$ by (p,Ψ_i,q) with $\Psi_i = \bigvee_{J \ni i} \psi_J$. Then $L(\mathscr{B}') = L(\mathscr{B})$, and one can continue with the usual powerset construction.

Concerning the second part of the Lemma, for a given $\mathscr{M}$-$\mathscr{L}$-recognizable $U \subseteq (M^n)^*$, the construction of an $\mathscr{M}$-$\mathscr{L}$-Büchi automaton recognizing U^ω can be done in a straightforward way by isolating the initial state such that it has no incoming transitions and for each transition from a state q to some state in F, adding a transition from q to the initial state over the same letter, where the initial state will be the only final state in the new automaton. For the concatenation $U \cdot K$, we again follow a well-known idea by composing the two automata with additional transitions to cross over from one to the other at the appropriate positions. ■

The basic decidability result on $\mathscr{M}$-$\mathscr{L}$-automata is the following. We state it for both kinds of automata:

Proposition 2 *If the $\mathscr{L}$-theory of $\mathscr{M}$ is decidable, then the nonemptiness problem for $\mathscr{M}$-$\mathscr{L}$-automata on finite words as well as for $\mathscr{M}$-$\mathscr{L}$-Büchi automata is decidable.*

Proof For both kinds of $\mathscr{M}$-$\mathscr{L}$-automata, we have to determine whether there exists a word which is the label of a finite successful run. As a preparation, we have to check for each of the finitely many transitions $(p,\varphi(x_1,\dots,x_n),q) \in \Delta$ whether it is “useful”, i.e., whether there is an input letter $\overline{m} \in M^n$ satisfying φ. This is done by invoking decidability of the $\mathscr{L}$-theory of $\mathscr{M}$, namely by checking whether $\mathscr{M} \models \exists x_1 \dots \exists x_n \varphi(x_1,\dots,x_n)$. Now one considers the directed graph (Q,R) where $(p,q) \in R$ if there is a useful transition from p to q. For an $\mathscr{M}$-$\mathscr{L}$-automaton over finite words, it remains to check whether in (Q,R) there is a path from q_0 to F; for an $\mathscr{M}$-$\mathscr{L}$-Büchi automaton one verifies whether in (Q,R) there is a path from q_0 to a strongly connected component containing a state from F. ■

We now show basic closure properties of $\mathscr{M}$-$\mathscr{L}$-Büchi automata.

Lemma 3 *If the $\mathscr{L}$-theory of $\mathscr{M}$ is decidable, the class of $\mathscr{M}$-$\mathscr{L}$-Büchi-recognizable ω-languages is effectively closed under union, projection, and complementation.*

Proof For union and projection the same construction as in Lemma 1 works. We sketch the construction for complementation, using the original approach of Büchi [3].

Let $\mathscr{B} = (Q, M^n, q_0, \Delta, F)$ be an $\mathscr{M}$-$\mathscr{L}$-Büchi automaton. We introduce an equivalence relation over finite M^n-words such that $(M^n)^\omega \setminus L(\mathscr{B})$ is representable as a finite union of sets $U \cdot V^\omega$ with $\mathscr{M}$-$\mathscr{L}$-recognizable sets $U, V \subseteq (M^n)^*$. By Lemma 1, this suffices to show Büchi recognizability of $(M^n)^\omega \setminus L(\mathscr{B})$.

The desired equivalence relation is defined in terms of *transition profiles*. We write for a finite word $u \in (M^n)^*$ and $p, q \in Q$:

- $\mathscr{B} : p \xrightarrow{u} q$ if there is a run on u from p to q in $\mathscr{B}$,
- $\mathscr{B} : p \xrightarrow[F]{u} q$ if there is a run on u from p to q in $\mathscr{B}$ that visits an accepting state from F.

A transition profile $\tau = tp(u)$ is then given by two sets $I_{tp(u)}$, $J_{tp(u)}$ of pairs of states, $I_{tp(u)}$ containing those pairs (p,q) where $\mathscr{B} : p \xrightarrow{u} q$, and $J_{tp(u)}$ containing those pairs (p,q) where $\mathscr{B} : p \xrightarrow[F]{u} q$. Two words u, v are called $\mathscr{B}$-equivalent, written $u \sim_{\mathscr{B}} v$, if $tp(u) = tp(v)$. This equivalence relation is of finite index: For this, note that each equivalence class (i.e., a language U_τ for a type τ) is a Boolean combination of the $\mathscr{M}$-$\mathscr{L}$-recognizable languages $U_{pq} = \{u \mid \mathscr{B} : p \xrightarrow{u} q\}$, $U'_{pq} = \{u \mid \mathscr{B} : p \xrightarrow[F]{u} q\}$, in fact, we have

$$U_\tau = \bigcap_{(p,q)\in I_\tau} U_{pq} \cap \bigcap_{(p,q)\notin I_\tau} \overline{U_{pq}} \cap \bigcap_{(p,q)\in J_\tau} U'_{pq} \cap \bigcap_{(p,q)\notin J_\tau} \overline{U'_{pq}}.$$

Since the set of pairs (p,q) is finite, we get only finitely many equivalence classes. Moreover, by Lemma 1 and Proposition 2, we can compute those U_τ which are nonempty and hence obtain an effective presentation of the equivalence classes in terms of the corresponding finite sets I_τ, J_τ.

We identify the equivalence classes with the transition profiles and denote the set of these transition profiles of $\mathscr{B}$ by $TP_{\mathscr{B}}$.

The following “saturation property” is now immediate:

Lemma 4 *For any $\sim_{\mathscr{B}}$-equivalence classes U, V, the ω-language $U \cdot V^\omega$ is either contained in $L(\mathscr{B})$ or in its complement.*

It remains to show that any ω-word over M^n belongs to some set $U \cdot V^\omega$ where U, V are $\sim_{\mathscr{B}}$-classes. For this we use the transition profiles as “colors” of segments $\alpha[i,j]$ for $i, j \in \mathbb{N}$. By Ramsey’s Infinity Lemma [11] there is for any α and any Büchi automaton $\mathscr{B}$ a pair of transition profiles τ_0, τ from $TP_{\mathscr{B}}$ and an infinite set $I = \{i_0 < i_1 < i_2 < \ldots\}$ such that

$$tp(\alpha[0, i_0 - 1]) = \tau_0, \;\; tp(\alpha[i_j, i_{j+1} - 1]) = \tau \text{ for } j \geq 0.$$

This shows that $\alpha \in U_{\tau_0} \cdot U_\tau^\omega$, where U_{τ_0}, U_τ denote the equivalence classes of $\sim_{\mathscr{B}}$ corresponding to τ_0 resp. τ. Let

$$NTP_{\mathscr{B}} = \{(\tau_0, \tau) \in TP_{\mathscr{B}}^2 \mid U_{\tau_0} \cdot U_\tau^\omega \cap L(\mathscr{B}) = \emptyset\}$$

Again, by decidability of the $\mathscr{L}$-theory of $\mathscr{M}$, this set is computable. Then

$$(M^n)^\omega \setminus L(\mathscr{B}) = \bigcup_{(\tau_0,\tau)\in NTP} U_{\tau_0} U_\tau^\omega.$$

■

As a consequence of Lemma 1 and Lemma 3 we obtain the following result.

Proposition 5 *If the $\mathscr{L}$-theory of $\mathscr{M}$ is decidable, the inclusion problem and the equivalence problem for $\mathscr{M}$-$\mathscr{L}$-Büchi recognizable languages are decidable.*

After these preparations, one can easily infer an equivalence between $\mathscr{M}$-$\mathscr{L}$-Büchi automata and $\mathscr{M}$-$\mathscr{L}$-MSO.

Remark 6 Let $\mathscr{B}$ be an $\mathscr{M}$-$\mathscr{L}$-Büchi automaton, then there exists an $\mathscr{M}$-$\mathscr{L}$-MSO sentence ψ with $L(\mathscr{B}) = L(\psi)$.

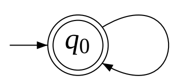
Again, the construction of an $\mathscr{M}$-$\mathscr{L}$-MSO formula describing a successful run of a given $\mathscr{M}$-$\mathscr{L}$-Büchi automaton $\mathscr{B}$ is a straightforward adaption of the well-known proof ([17]). The only modification occurs in the formulas describing the transitions of $\mathscr{B}$: for a transition (p,φ,q), one uses the predicates $P_\varphi(x)$ as introduced above in the definition of $\mathscr{M}$-$\mathscr{L}$-MSO.

Let us turn to the translation from $\mathscr{M}$-$\mathscr{L}$-MSO sentences to $\mathscr{M}$-$\mathscr{L}$-Büchi automata.

Proposition 7 *Let ψ be an $\mathscr{M}$-$\mathscr{L}$-MSO sentence, then there exists an $\mathscr{M}$-$\mathscr{L}$-Büchi-automaton $\mathscr{B}$ with $L(\psi) = L(\mathscr{B})$.*

Proof We first modify $\mathscr{M}$-$\mathscr{L}$-MSO to the expressively equivalent formalism of $\mathscr{M}$-$\mathscr{L}$-MSO_0-formulas in complete analogy to the definition of chain_0 logic in Section 2. We proceed by induction over MSO_0-formulas.

For the induction basis, we consider the atomic formulas $X_i \subseteq X_j$, $\text{Sing}(X_i)$, $\text{Succ}(X_i,X_j)$, $X_i \preceq X_j$, and $X_i \subseteq P_\varphi$ and specify $\mathscr{M}$-$\mathscr{L}$-Büchi automata that recognize the sets of ω-words defined by these formulas. To exemplify, we give the automaton for $X_i \subseteq P_\varphi$, which checks that when the i-th component is 1, the letter vector satisfies the $\mathscr{M}$-$\mathscr{L}$-formula φ, which defines the letter predicate P_φ.



For the induction step, we consider the connectives $\vee$ and $\neg$, as well as the existential quantifier $\exists$. Here, we can exploit the closure properties of $\mathscr{M}$-$\mathscr{L}$-Büchi automata from Lemma 3, and employ the constructions for the union, complementation, and projection, respectively. ■

As a relation $R \subseteq (M^\omega)^n$ is representable by a convolution as an ω-word over M^n, Remark 6 and Proposition 7 yield the following result.

Theorem 8 *A relation $R \subseteq (M^\omega)^n$ with $n \geq 1$ of ω-words is $\mathscr{M}$-$\mathscr{L}$-MSO definable iff it is $\mathscr{M}$-$\mathscr{L}$-Büchi-recognizable. The transformation in both directions is effective.*

As a consequence of the $\mathscr{M}$-$\mathscr{L}$-Büchi theory, we obtain that satisfiability and equivalence of $\mathscr{M}$-$\mathscr{L}$-MSO-formulas over models from M^ω are decidable if the $\mathscr{L}$-theory of the structure $\mathscr{M}$ is decidable.

3.2 Strong $\mathscr{M}$-$\mathscr{L}$-Büchi automata

In the second part of this section, we extend – as far as possible – the techniques and results to a slightly stronger model of Büchi automaton. While the Büchi automata above are appropriate for treating the structures $\mathscr{M}_E^\#$, a stronger model is motivated by the study of strong tree iterations $\mathscr{M}_E^*$ in which the clone predicate enters. Recall that it allows to single out those elements of M^* which are of the form $u\,m\,m$. Thus, when reading a "letter" m along a path, we need to incorporate the feature to "remember" whether this current input letter m coincides with the previous one.

We define the notion of *strong $\mathscr{M}$-$\mathscr{L}$-Büchi automaton* over n-tuple input letters (i.e., with input alphabet M^n, M being the domain of $\mathscr{M}$). The format is the same as for standard Büchi automata over M^n as mentioned above, except for the transitions. For each state pair (p,q) the possible transitions are defined by a formula $\varphi_{pq}(x_1,\ldots,x_n,y_1,\ldots,y_n)$ – or, in the special case of an initial transition, by a formula $\varphi_{q_0q}(x_1,\ldots x_n)$. Starting with the latter case, the automaton can proceed from q_0 to q with input letter $(m_1,\ldots,m_n)$ if $\mathscr{M}\models\varphi[m_1,\ldots,m_n]$. For a transition of the first case, in which a previous input letter exists and is $(m_1^-,\ldots,m_n^-)$, the automaton can move from p to q if $\mathscr{M}\models\phi[m_1,\ldots,m_n,m_1^-,\ldots,m_n^-]$. All other notions are copied from the case of (standard) $\mathscr{M}$-$\mathscr{L}$-Büchi automata.

We can reprove the basic decidability and closure properties only under rather radical restrictions, namely just for the logic $\mathscr{L}$ = MSO and for the case of input letters from M (rather than n-tuples of such letters). We only give a rough outline; in the present paper we do not apply these automata to chain logic over tree structures.

First let us state the basic decidability result.

Lemma 9 *If the MSO-theory of $\mathscr{M}$ is decidable, the emptiness problem for strong $\mathscr{M}$-MSO-Büchi automata over M is decidable.*

Proof The proof of this lemma can either be given directly, or by invoking the above-mentioned Muchnik-Walukiewicz result ([12, 19]). It states – under the assumption that the MSO-theory of $\mathscr{M}$ is decidable – that the MSO-theory of $\mathscr{M}^*$ is decidable. The nonemptiness of a strong Büchi automaton over M can be decided by checking existence of a suitable path through $\mathscr{M}^*$. ■

Lemma 10 *If the MSO-theory of $\mathscr{M}$ is decidable, the class of ω-languages recognized by strong $\mathscr{M}$-MSO-Büchi automata over M is effectively closed under the Boolean operations and definable projections $p: M\to M$.*

Proof This claim is shown in precise analogy to the case of standard $\mathscr{M}$-$\mathscr{L}$-Büchi automata (and we skip here the repetition of proofs), except for the closure under complement. Here we describe the necessary modifications.

The approach is the same as for the standard case, i.e., via Büchi's original method involving finite colorings and Ramsey's Theorem. However, the coloring of a segment of an ω-word over the alphabet M^n, i.e., the transition profile, is defined differently. Given a strong Büchi automaton $\mathscr{A}$, the "strong transition profile" of the segment $\alpha[i,j]$ of an ω-word α refers also to the last previous letter $\alpha(i-1)$ if $i>0$. This extra context information is needed in order to capture the clone predicate on the n components of α, and we define the transition profile of a segment relative to this context information within α. So an appropriate notation for a strong transition profile is $tp_\alpha[i,j])$ rather than $tp(u)$. Such profiles, however, are of the same type as the previously defined profiles (namely, presented as two sets of pairs of states). The transition profile of a segment $\alpha[i,j]$ is fixed from the state pairs (p,q) that allow a run of the automaton from p to q (respectively, a run from p to q via a final state), where in the first move the letter $\alpha(i-1)$ is used. (This condition is dropped for the case $i=0$.)

There is, of course, a definite conceptual difference to the usual coloring of segments in terms of standard transition profiles: There, one may concatenate any sequence of segments (for given transition profiles) to obtain a new composed segment whose transition profile is induced by the given ones. In the new setting, the composition of segments u and v only works when the clone information on the last letter of u agrees with the first letter of v. However, this does not affect the argument in Büchi's complementation proof: Here we only need that for any *given* α one can obtain a sequence $i_0 < i_1 < \dots$ such that all segments $\alpha[i_j, i_{j+1}-1]$ share the same transition profile, and that for such a sequence, the transition profiles of $\alpha[0, i_0 - 1]$ and of $\alpha[i_0, i_1 - 1]$ determine α either to be accepted of not to be accepted by the Büchi automaton.

Also the sets $U_{\tau_0} \cdot U_\tau^\omega$ can be used as before when defined properly: Such a set is not obtained by freely concatenating a segment $u \in U_{\tau_0}$ and a sequence of segments from U_τ; rather, it is the set

$$U_{\tau_0} \cdot U_\tau^\omega = \{\alpha \mid \exists i_0, i_1, \dots (0 < i_0 < i_1 < \dots \wedge tp_\alpha[0, i_0 - 1] = \tau_0 \wedge tp_\alpha[i_j, i_{j+1} - 1] = \tau \text{ for } j = 0, 1, \dots)\}$$

The effective presentation of the complement of $L(\mathscr{A})$ is now completed as in the preceding subsection for $\mathscr{M}$-$\mathscr{L}$-Büchi automata. ■

In Section 5 below we shall see that these results fail for the case of an infinite alphabet M^n with infinite M and $n > 1$.

4 Weak Tree Iterations

In this section, we want to show that for the weak tree iteration with equal level relation, the chain theory with $\mathscr{L}$ on siblings is decidable if the $\mathscr{L}$-theory of $\mathscr{M}$ is.

With the preparations of Section 3, we will establish a reduction from chain logic formulas over tree models to $\mathscr{M}$-$\mathscr{L}$-MSO over ω-sequences (and then to Büchi automata).

To avoid heavy notation, we employ chain$_0$ logic as introduced in Section 2, and provide the following construction. Recall that for a chain c in $\mathscr{M}_E^\#$, the object $\hat{c}$ is a pair of sequences over M coding the path underlying the chain c, respectively the membership of nodes of this path in c.

Lemma 11 *For any chain$_0$-formula $\varphi(X_1, \dots, X_n)$ over $\mathscr{M}_E^\# = (M^*, S, \preceq, R_1^*, \dots, R_k^*, E)$ with $\mathscr{L}$ on siblings, one can construct an $\mathscr{M}$-$\mathscr{L}$-MSO-formula $\varphi'(Y_1, Z_1, \dots, Y_n, Z_n)$ interpreted in ω-words over M^{2n} such that for all chains $c_1, \dots, c_n$ we have:*

$$\mathscr{M}_E^\# \models \varphi[c_1, \dots, c_n]$$
$$\text{if and only if } \underline{\langle \hat{c}_1, \dots, \hat{c}_n \rangle} \models \varphi'(Y_1, Z_1, \dots, Y_n, Z_n).$$

Proof We proceed by induction over the structure of chain$_0$-formulas with $\mathscr{L}$ on siblings over $\mathscr{M}_E^\#$.

For the induction basis we have to consider the atomic formulas, namely of the form $\mathrm{Sing}(X)$, $X_i \subseteq X_j$, $X_i \preceq X_j$, $R_i^*(X_1, \dots, X_k)$, $E(X_i, X_j)$, and also the $\mathscr{L}$-formulas $\gamma(x_{i_1}, \dots, x_{i_\ell})$.

As a first example, we present the translation into $\mathscr{M}$-$\mathscr{L}$-MSO-formulas for the formula $\varphi(X) = \mathrm{Sing}(X)$: Given the encoding $\hat{c} = (\alpha, \beta)$ of a chain c, the formula $\varphi'_{\mathrm{Sing}}(X)$ has to express that β indicates membership in c exactly once. Thus, we obtain $\varphi'_{\mathrm{Sing}}(Y, Z) = \exists s\big(Z(s) \wedge \forall t(t \neq s \to \neg Z(s))\big)$.

For the case of an $\mathscr{L}$-formula $\gamma(x_{i_1}, \dots, x_{i_\ell})$, we capture $x_{i_1}, \dots, x_{i_\ell}$ by corresponding singletons $X_{i_1}, \dots, X_{i_\ell}$, and these in turn by pairs $(Y_{i_1}, Z_{i_1}), \dots, (Y_{i_\ell}, Z_{i_\ell})$ consisting of a path $Y_{i_j} \in M^\omega$ and a singleton set indicator $Z_{i_j} \subseteq \{0,1\}^\omega$ each. We have to define a corresponding predicate $P_\gamma \subseteq ((M \times \{0,1\})^n)^\omega$ by an $\mathcal{M}$-$\mathcal{L}$-MSO-formula that expresses in terms of the Y_{i_j}, Z_{i_j} that there is a common S-predecessor z of the elements x_{i_j} and that the tuple $x_{i_1},\dots,x_{i_\ell}$ satisfies γ. In intuitive notation, we have

$$\varphi'_{P_\gamma}(Y_1,Z_1,\dots,Y_n,Z_n) = \bigwedge_{j=1}^{\ell} \text{“}(Y_{i_j},Z_{i_j}) \text{ is singleton containing } x_{i_j}\text{”} \wedge \exists z \bigwedge_{j=1}^{\ell} \text{“}S(z,x_{i_j})\text{”} \wedge \gamma(x_{i_1},\dots,x_{i_\ell})$$

In some more detail:

$$\bigwedge_{j=1}^{\ell} \varphi'_{\text{Sing}}(Y_{i_j},Z_{i_j}) \wedge \exists x_{i_1}\dots\exists x_{i_\ell}\exists s\big(Z_{i_j}(s)\wedge Y_{i_j}(s) = x_{i_j} \wedge \bigwedge_{j'\neq j} \forall t < s\,(Y_{i_j}(t) = Y_{i_{j'}}(t)) \wedge \gamma(x_{i_1},\dots,x_{i_\ell})\big)$$

The induction step then is straightforward, as $\mathcal{M}$-$\mathcal{L}$-MSO is closed under the Boolean operations and projection. ■

Thus, we obtain a reduction of the chain$_0$-theory with $\mathcal{L}$ on siblings of $\mathcal{M}_E^\#$ to the $\mathcal{M}$-$\mathcal{L}$-MSO theory, which with Theorem 8 is decidable if the $\mathcal{L}$-theory of $\mathcal{M}$ is decidable. This leaves us to conclude this section with the following theorem:

Theorem 12 *If the $\mathcal{L}$-theory of $\mathcal{M}$ is decidable, the chain-theory of $\mathcal{M}_E^\#$ with $\mathcal{L}$ on siblings is decidable.*

5 Undecidability Results

In the previous sections we showed decidability of the model-checking problem for chain logic with $\mathcal{L}$ on siblings over tree structures $\mathcal{M}_E^\#$, given a structure $\mathcal{M}$ with decidable $\mathcal{L}$-theory for some logic $\mathcal{L}$.

The first result of this section shows that this does not extend to strong tree iterations $\mathcal{M}_E^*$ (even if we confine ourselves to first-order logic in place of chain logic).

The second result shows another limitation to decidability: In the “horizontal dimension” of tree models, we may (in Theorem 12) use $\mathcal{L}$-quantifiers ranging over children of given nodes. We show that for the case $\mathcal{L} = \text{MSO}$ we lose decidability when the horizontal quantification is extended to an entire tree level. Here we get undecidability for the weak tree iteration.

For the first result we use a reduction to the termination problem of 2-counter machines (or 2-register machines). Such a machine M is given by a finite sequence

$$1 \text{ instr}_1;\dots;k-1 \text{ instr}_{k-1};k \text{ stop}$$

where each instruction instr$_j$ is of the form

- $\text{Inc}(X_1)$, $\text{Inc}(X_2)$ (increment the value of X_1, respectively X_2 by 1), or
- $\text{Dec}(X_1)$, $\text{Dec}(X_2)$ (similarly for decrement by 1, with the convention that a decrement of 0 is 0), or
- If $X_i = 0$ goto ℓ_1 else to ℓ_2 (where $i = 1,2$ and $1 \leq \ell_1,\ell_2 \leq k$, with the natural interpretation).

An M-configuration is a triple (ℓ,m,n), indicating that the ℓ-th instruction is to be executed and the values of X_1,X_2 are m,n, respectively. A terminating M-computation (for M as above) is a sequence $(\ell_0,m_0,n_0),\dots,(\ell_r,m_r,n_r)$ of M-configurations where in each step the update is done according to the instructions in M and the last instruction is the stop-instruction (formally: $\ell_r = k$). The termination problem for 2-counter machines asks to decide, for any given 2-counter machine M, whether there exists a terminating M-computation that starts with $(1,0,0)$ (abbreviated as $M:(1,0,0)\to$ stop). It is well-known that the termination problem for 2-counter machines is undecidable ([7]).

We turn to the model-checking problem over structures $\mathscr{M}_E^*$. We show undecidability when $\mathscr{M}$ is the structure $\mathscr{S} := (\mathbb{N}, Suc)$ (where Suc is successor).

Theorem 13 *The first-order theory of $\mathscr{S}_E^*$ with FO on siblings is undecidable.*

Proof For any 2-register machine M we construct an first-order formula φ_M with FO on siblings such that $M:(1,0,0)\to$ stop iff $\mathscr{S}_E^* \models \varphi_M$.

The idea is to code a computation $(\ell_0,m_0,n_0),\ldots,(\ell_r,m_r,n_r)$ by three finite paths of same length, one for each of the three components. Each of these paths (namely $\pi_0 = (\ell_0,\ldots,\ell_r), \pi_1 = (m_0,\ldots,m_r), \pi_2 = (n_0,\ldots,n_r)$) is determined by its last point in the tree structure $\mathscr{S}_E^*$, i.e., by a triple x_0, x_1, x_2 of $\mathscr{S}_E^*$-elements.

We use a formula which expresses

$$\exists x_0 \exists x_1 \exists x_2 (E(x_0,x_1) \wedge E(x_1,x_2) \wedge [x_0,x_1,x_2 \text{ code a terminating computation of } M]).$$

In order to obtain a formalization of the condition in squared brackets, we have to express

1. the initial condition that π_0 starts with the son 1 of the root and π_1,π_2 with the son 0 of the root,
2. the progress condition that for each $y_0 \prec x_0$ (giving an instruction number), the corresponding M-instruction is executed, which involves the vertex y_0 and the vertices $y_1 \prec x_1, y_2 \prec x_2$ on the same level as y_0 and their respective successors z_0,z_1,z_2 on π_0,π_1,π_2, respectively,
3. the termination condition that x_0 is the number k.

Accordingly, we can formalize the condition in squared brackets by a conjunction of three formulas φ_1, φ_2, φ_3 in the free variables x_0, x_1, x_2, making use of the (definable) tree successor relation S.

- The formula φ_1 expresses (in first-order logic with FO on siblings) for the root r of the tree model and those three S-successors y_0, y_1, y_2, where $y_0 \preceq x_0, y_1 \preceq x_1, y_2 \preceq x_2$, that y_0 is the number 1 and y_1, y_2 are the number 0 (of the model $\mathscr{S} = (\mathbb{N}, Suc)$).
- The formula φ_2 is of the form:

  "for all $y_0 \prec x_0$, $y_1 \prec x_1$, $y_2 \prec x_2$ with $E(y_0,y_1)$ and $E(y_0,y_2)$, there are tree-successors z_0, z_1, z_2 (i.e., with $S(y_0,z_0)$, $S(y_1,z_1)$, $S(y_2,z_2)$ with $z_0 \preceq x_0$, $z_1 \preceq x_1$, $z_2 \preceq x_2$) that represent the correct update of the configuration (y_0,y_1,y_2)."

  The condition on update is expressed by a disjunction over all program instructions; we present, as an example, the disjunction member for the statement "3 Inc(X_2)":

  $$\begin{aligned} y_0 \text{ is number 3 in } (\mathbb{N},Suc) \to z_0 \text{ is number 4 in } (\mathbb{N},Suc) \\ \wedge z_1 \text{ is the clone of } y_1 \wedge z_2 \text{ is the } Suc\text{-successor of the clone of } y_2. \end{aligned}$$

  It is easy to formalize this in first-order logic with FO on siblings, similarly for the Dec-instructions and the jump instructions.
- The formula φ_3 expresses the third condition and is clearly formalizable in first-order logic with FO on siblings. ∎

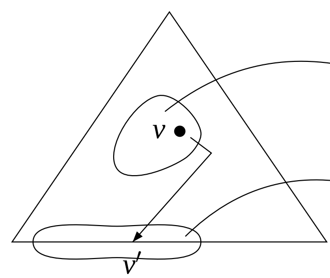


Figure 1: Coding an element of a set S by an element of an antichain A.

This result can also be stated in the framework of strong Büchi automata (or even strong automata on finite words) when the alphabet consists of pairs of natural numbers: With each 2-register machine M one associates a strong $\mathscr{S}$-MSO-automaton $\mathscr{A}_M$ over $\mathbb{N}^2$ which accepts an input word $(m_0,n_0)\ldots(m_r,n_r)$ if this represents the sequence of register values of a terminating computation of M; the existence of an appropriate sequence of instruction numbers (from $\{1,\ldots,k\}$) can be expressed by a block $\exists X_1 \ldots \exists X_k$ of MSO-quantifiers. (In fact, weak MSO-quantifiers suffice.)

Let us turn to the second undecidability result. We shall confine ourselves to the simplest setting, where the structure $\mathscr{M}$ is just $(\{0,1\},\{0\},\{1\})$, i.e., $\mathscr{M}_E^{\#}$ and $\mathscr{M}_E^{*}$ are *both* the binary tree with equal level relation (see also [18]).

Theorem 14 *The chain theory of the binary tree with equal level relation and MSO on tree levels is undecidable.*

Proof We use an idea of [9] that allows to code a tuple of finite sets of the binary tree up to (and excluding) level L by a tuple of subsets of level L itself. In other words, we code a subset S of tree nodes before level L by an "antichain" A which is a subset of the level L (see Figure 1).

We simply map a vertex v (before level L) to the unique vertex $v' \in L$ which belongs to $v10^*$ (i.e., belongs to the leftmost path from the right successor of v; see again Figure 1). The map $v \mapsto v'$ is injective and definable in chain logic (even in FO-logic), given the level L. Moreover, it is easy to see that the relations of being left or right successor in the tree are translated to FO-definable relations over the level L under consideration.

Using this coding, an existential quantifier over finite sets in the binary tree is captured by an existential quantifier over subsets of an appropriate level of the tree (namely, of a level that is beyond all maximal elements of the finite set under consideration).

Thus, the weak MSO-theory of the binary tree with E is interpretable in the FO-theory of the binary tree $(\{0,1\}^*,S_0,S_1,\preceq,E)$ with E and with MSO restricted to levels.

Since the weak MSO-theory of the binary tree with E is undecidable (see e.g. [15]), we obtain the claim. ■

6 Conclusion

In this work, we outlined a theory of generalized Büchi automata over infinite alphabets. These alphabets are represented by relational structures $\mathscr{M}$, the transitions being specified by formulas of a logic $\mathscr{L}$ over $\mathscr{M}$. In this setting of $\mathscr{M}$-$\mathscr{L}$-Büchi automata (which only slightly generalizes that of [1]), the nonemptiness problem becomes decidable if the $\mathscr{L}$-theory of $\mathscr{M}$ is. An extended model of strong $\mathscr{M}$-$\mathscr{L}$-Büchi automata was introduced in which a transition via an $\mathscr{M}$-input may depend on the previous $\mathscr{M}$

input. Here an essential difference appears between the cases where input letters are from M and where input letters are in M^n for $n > 1$.

We applied this theory to show that the chain logic theory of the weak tree iteration $\mathscr{M}^{\#}_{E}$ of $\mathscr{M}$ (with $\mathscr{L}$ chosen as above) is decidable where the equal level relation is adjoined, and quantifications of $\mathscr{L}$ over siblings of the tree model are allowed. On the other hand, we showed limits for generalization. For example, we showed undecidability for the corresponding theory of the strong tree iteration when the underlying model is the successor structure of the natural numbers.

Several problems are raised by this study. Since the logics considered here all have nonelementary complexity, it may be interesting to set up fragments and "dialects" (e.g. in temporal logics) of chain logic where the complexity is better. Also, it seems that variants of the model of strong (Büchi-) automaton should be studied in more depth, for instance by an integration with the theory of automata over "data words" as developed in [8, 2, 4].

References

[1] A. Bès (2008): *An Application of the Feferman-Vaught Theorem to Automata and Logics for Words over an Infinite Alphabet*. *Logical Methods in Computer Science* 4, pp. 1–23.

[2] M. Bojanczyk, A. Muscholl, T. Schwentick, L. Segoufin & C. David (2006): *Two-variable logic on words with data*. In: *Proceedings of the 21th IEEE Symposium on Logic in Computer Science (LICS '06)*, IEEE Computer Society, pp. 7–16.

[3] J.R. Büchi (1962): *On a decision method in restricted second order arithmetic*. In: *Logic, Methodology and Philosophy of Science: Proceedings of the 1960 International Congress*, Stanford Univ. Press, pp. 1–11.

[4] C. Choffrut & S. Grigorieff (2009): *Finite n-tape automata over possibly infinite alphabets: Extending a theorem of Eilenberg et al*. *Theor. Comput. Sci.* 410(1), pp. 16–34.

[5] H.-D. Ebbinghaus, J. Flum & W. Thomas (2007): *Einführung in die mathematische Logik*, 5 edition. Spektrum Akademischer Verlag, Heidelberg.

[6] D. Kuske & M. Lohrey (2006): *Monadic chain logic over iterations and applications to pushdown systems*. In: *Logic in Computer Science, 2006*, IEEE Computer Society, pp. 91–100.

[7] M. Minsky (1967): *Computation: finite and infinite machines*. Prentice-Hall.

[8] F. Neven, T. Schwentick & V. Vianu (2004): *Finite state machines for strings over infinite alphabets*. *ACM Trans. Comput. Log.* 5(3), pp. 403–435.

[9] A. Potthoff & W. Thomas (1993): *Regular tree languages without unary symbols are star-free*. In: *Proceedings of Fundamentals of Computation Theory, FCT '93*, *LNCS* 710, Springer, pp. 396–405.

[10] M.O. Rabin (1969): *Decidability of second-order theories and automata on infinite trees*. *Trans. Amer. Math. Soc* 141(1), pp. 1–35.

[11] F.P. Ramsey (1930): *On a problem of formal logic*. *Proceedings of the London Mathematical Society* 2(1), p. 264.

[12] A.L. Semenov (1984): *Decidability of monadic theories*. In: *Proceedings of Mathematical Foundations of Computer Science, MFCS '84*, *LNCS* 176, Springer, pp. 162–175.

[13] S. Shelah (1975): *The monadic theory of order*. *Annals of Mathematics* 102, pp. 379–419.

[14] J. Stupp (1975): *The lattice-model is recursive in the original model*. Technical Report, Institute of Mathematics, The Hebrew University, Jerusalem.

[15] W. Thomas (1990): *Automata on Infinite Objects*. In J. van Leeuwen, editor: *Handbook of Theoretical Computer Science: Formal Models and Sematics*, B, Elsevier and MIT Press, pp. 133–192.

[16] W. Thomas (1990): *Infinite trees and automaton definable relations over omega-words*. In: *Proceedings of the 7th Annual Symposium on Theoretical Aspects of Computer Science, STACS '90*, Springer, pp. 263–277.

[17] W. Thomas (1997): *Languages, automata, and logic*. In G. Rozenberg & A. Salomaa, editors: *Handbook of formal languages*, 3, Springer, New York, pp. 389–455.

[18] W. Thomas (2009): *Path logics with synchronization*. In K. Lodaya, M. Mukund & R. Ramanujam, editors: *Perspectives in Concurrency Theory*, IARCS-Universities, Universities Press, pp. 469–481.

[19] I. Walukiewicz (2002): *Monadic second-order logic on tree-like structures*. *Theoretical Computer Science* 275(1-2), pp. 311–346.