Isomorphisms of scattered automatic linear orders

Dietrich Kuske

Institut für Theoretische Informatik, TU Ilmenau, Germany

Abstract

We prove that the isomorphism of scattered tree-automatic linear orders as well as the existence of automorphisms of scattered word-automatic linear orders are undecidable. For the existence of automatic automorphisms of word- or tree-automatic linear orders, we determine the exact level of undecidability in the arithmetical hierarchy.

1. Introduction

Automatic structures form a class of computable structures with much better algorithmic properties: while, due to Rice's theorem, nothing is decidable about a computable structure (given as a tuple of Turing machines), validity of first-order sentences is decidable in automatic structures (given as a tuple of finite automata). This property of automatic structures was first observed and exploited in concrete settings by Büchi, by Elgot [11], and by Epstein et al. [12]. Hodgson [14] attempted a uniform treatment, but the systematic study really started with the work by Khoussainov and Nerode [19] and by Blumensath and Grädel [3, 4]. Over the last decade, a fair amount of results have been obtained, see e.g. the surveys [33, 2] as well as the list of open questions [20], for very recent results not covered by the mentioned articles, see e.g. [18, 5, 10, 16, 15, 17].

A rather basic question about two automatic structures asks whether they are isomorphic. For word-automatic ordinals and Boolean algebras, this problem was shown to be decidable via a characterisation of the word-automatic members of these classes of structures [8, 22, 21]. The same applies to many classes of unary automatic structures (i.e., structures over a unary alphabet) [26]. On the other hand, already Blumensath and Grädel [4] observed that this problem is undecidable in general. In [21], it is shown that the isomorphism problem is Σ^1_1-complete; a direct interpretation yields the same result for many natural classes of automatic structures [31]. Rubin [32] shows that the isomorphism problem for locally finite graphs is complete for Π^0_3. Recently, Miasknikov and Šunić proved that the isomomorphism of Cayley automatic groups is undecidable [29]. In [25], we show in particular that also the isomorphism problems of order trees and of linear orders are Σ^1_1-complete. For the handling of linear orders, our arguments rely heavily on "shuffle sums". Consequently, we construct linear orders that contain a copy of the rational line (a linear order not containing the rational line is called scattered, i.e., our result is shown for non-scattered linear orders). This is unavoidable since we also show that the isomorphism problem for word-automatic scattered linear orders is reducible to true arithmetic (i.e., the first-order theory of $(\mathbb{N}; +, \cdot)$) and therefore much "simpler" than the isomorphism problem for arbitrary linear orders (cf. [17] for further evidence that scattered automatic linear orders behave better than general automatic linear orders). But it is still conceivable that the isomorphism problem for scattered linear orders is decidable.

In this paper, we deal with automatic scattered linear orders. In particular, we prove the following three results:

(1) There is a scattered linear order whose set of tree-automatic presentations is Π^0_1-hard (Theorem 4.11). Hence also the isomorphism problem for tree-automatic scattered linear orders is Π^0_1-hard (Corollary 4.12).

(2) The existence of a non-trivial automorphism of a word-automatic scattered linear order is Σ^0_1-hard (Theorem 3.4). The existence of an automatic non-trivial automorphism is Σ^0_1-complete.

For regular languages ordered lexicographically, the existence of a non-trivial automorphism is known to be decidable in polynomial time (provided the regular language is given by a deterministic finite automaton, [27]), but we show it to be undecidable for contextfree languages (Theorem 3.7).

(3) The existence of a non-trivial automorphism of a tree-automatic scattered linear order is Σ^0_2-hard (Theorem 4.17).

The proof of (2) uses, similarly to [25], an encoding of polynomials but avoids the use of shuffle sums. The technique for proving (1) and (3) is genuinely new: One can understand a weighted automaton over the semiring $(\mathbb{N} \cup \{-\infty\}; \max, +)$ as a classical automaton with a partition of the set of transitions into two sets T_0 and T_1. The behavior of such a weighted automaton assigns numbers to words w, namely the maximal number of transitions from T_1 in an accepting run on the word w. Krob [23] showed that the equivalence problem for such weighted automata is Π^0_1-complete. This result was sharpened in [9] where it is shown that there is a single weighted automaton $\mathcal{A}$ such that the set of weighted automata equivalent to $\mathcal{A}$ is Π^0_1-complete. Based on ideas from [1], Section 4.1.3 contains a simplified proof of this sharpened result and a new sharpening, namely the existence of two fixed weighted automata such that it is undecidable whether they behave the same on a given regular language. A closer analysis of this proof, together with the techniques for proving (1) and (2), finally yields (3).

These results show that the existence of isomorphisms and of automorphisms is non-trivial for scattered linear orders that are described by word- and tree-automata, resp.

2. Preliminaries

2.1. Tree- and word-automatic structures

Let Γ be some alphabet. A Γ-*tree* or just a *tree* is a finite partial mapping $t\colon \{0,1\}^* \dashrightarrow \Gamma$ such that $uv \in \mathrm{dom}(t)$ implies $u \in \mathrm{dom}(t)$, and $u1 \in \mathrm{dom}(t)$ implies $u0 \in \mathrm{dom}(t)$ (note that we allow the empty tree $\emptyset$ with $\mathrm{dom}(\emptyset) = \emptyset$). A *(bottom up) tree-automaton* is a tuple $\mathcal{A} = (Q, \iota, \Delta, F)$ where Q is a finite set of *states*, ι is the initial state, $\Delta \subseteq Q \times \Gamma \times Q^2$ is the *transition relation*, and $F \subseteq Q$ is the set of *final states*. A *run* of the tree-automaton $\mathcal{A}$ on the tree t is a mapping $\rho\colon \mathrm{dom}(t) \to Q$ such that

$$(\rho(u), t(u), \rho'(u0), \rho'(u1)) \in \Delta \text{ with } \rho'(v) = \begin{cases} \rho(v) & \text{for } v \in \mathrm{dom}(t) \\ \iota & \text{otherwise} \end{cases}$$

holds for all $u \in \mathrm{dom}(t)$. The run ρ is *accepting* if $\rho(\varepsilon) \in F$. The *language of the tree-automaton* $\mathcal{A}$ is the set $L(\mathcal{A})$ of all trees t that admit an accepting run of $\mathcal{A}$ on t. A set L of trees is *regular* if there exists a tree-automaton $\mathcal{A}$ with $L(\mathcal{A}) = L$.

It is convenient to understand a *word* as a tree t with $\mathrm{dom}(t) \subseteq 0^*$ (then $t(\varepsilon)$ is the first letter of the word). Nevertheless, we will use standard notation for words like uv for the concatenation or ε for the empty word. A *word-automaton* is a tree-automaton $\mathcal{A} = (Q, \iota, \Delta, F)$ with

$$(q, a, p_0, p_1) \in \Delta \implies p_1 = \iota \text{ and } q \neq \iota\,.$$

This condition ensures that word-automata accept words, only.

Let $t_1, \dots, t_n$ be trees and let $\# \notin \Gamma$. Then $\Gamma_\# = \Gamma \cup \{\#\}$ and the *convolution* $\otimes(t_1, t_2, \dots, t_n)$ or $t_1 \otimes t_2 \otimes \dots \otimes t_n$ is the $\Gamma_\#^n$-tree t with $\mathrm{dom}(t) = \bigcup_{1 \le i \le n} \mathrm{dom}(t)$ and

$$t(u) = (t_1'(u), t_2'(u), \dots, t_n'(u)) \text{ with } t_i'(u) = \begin{cases} t_i(u) & \text{if } u \in \mathrm{dom}(t_i) \\ \# & \text{otherwise.} \end{cases}$$

Note that the convolution of a tuple of words is a word, again. For an n-ary relation R on the set of all trees, we write $R^\otimes$ for the set of convolutions $\otimes(t_1, \dots, t_n)$ with $(t_1, \dots, t_n) \in R$. A relation R on trees is *automatic* if $R^\otimes$ is a regular tree-language.

A relational structure $\mathcal{S} = (L; R_1, \dots, R_k)$ is *tree-automatic* if the tree-languages L and $R_i^\otimes$ for $1 \le i \le k$ are regular; it is *word-automatic* if, in addition, L is a word-language. A tuple of tree-automata accepting L and $R_i^\otimes$ for $1 \le i \le k$ is called a *tree- or word-automatic presentation* of the structure $\mathcal{S}$.

2.2. Linear orders

For words u and v, we write $u \le_{\mathrm{pref}} v$ if u is a prefix of v. Let Γ be some set linearly ordered by $\le$. Then $\le_{\mathrm{lex}}$ denotes the lexicographic order on the set of words Γ^*: $u \le_{\mathrm{lex}} v$ if $u \le_{\mathrm{pref}} v$ or there are $x, y, z \in \Gamma^*$, $a, b \in \Gamma$ with $u = xay$, $v = xbz$, and $a < b$. From the lexicographic order on Γ^*, we derive a linear order (denoted $\le_{\mathrm{lex}}^2$) on the set $\Gamma^* \otimes \Gamma^*$ of convolutions of words by

$$u \otimes v \le_{\mathrm{lex}}^2 u' \otimes v' :\Leftrightarrow u <_{\mathrm{lex}} u' \text{ or } u = u', v \le_{\mathrm{lex}} v'\,.$$

By $\le_{\mathrm{llex}}$, we denote the *length-lexicographic order* defined by $u \le_{\mathrm{llex}} v$ if $|u| < |v|$ or $|u| = |v|$ and $u \le_{\mathrm{lex}} v$. We next extend this linear order $\le_{\mathrm{llex}}$ to trees. Let t be a tree. Then $t{\restriction_{0^*}}$ (more precisely, $t{\restriction_{(0^* \cap \mathrm{dom}(t))}}$) is a word that can be understood as the "main branch" of the tree t. For $u \in \{0,1\}^*$, let $t|_u$ denote the subtree of t rooted at u (i.e., $\mathrm{dom}(t|_u) = \{v \mid uv \in \mathrm{dom}(t)\}$ and $t|_u(v) = t(uv)$ for $u \in \mathrm{dom}(t|_u)$. In particular, $t|_u = \emptyset$ for $u \notin \mathrm{dom}(t)$). Furthermore, $\tau(t)$ is the tuple of "side trees" of t, namely

$$\tau(t) = (t|_{0^i 1})_{0^i \in \mathrm{dom}(t)}\,.$$

We now define the extension $\le_{\mathrm{trees}}$ of $\le_{\mathrm{llex}}$ to trees setting $s <_{\mathrm{trees}} t$ if and only if

- $s{\restriction_{0^*}} <_{\mathrm{llex}} t{\restriction_{0^*}}$ or
- $s{\restriction_{0^*}} = t{\restriction_{0^*}}$ and there exists i such that $s|_{0^j 1} = t|_{0^j 1}$ for all $0 \le j < i$ and $s|_{0^i 1} <_{\mathrm{trees}} t|_{0^i 1}$.

In other words, we first compare the main branches of the trees s and t length-lexicographically and, if they are equal, compare the tuples $\tau(s)$ and $\tau(t)$ (length-)lexicographically (based on the extension $\leq_{\text{trees}}$ of the length-lexicographic order to trees). Since the "side trees" $t|_{0^j1}$ of any tree t are properly smaller than the tree itself, the relation $\leq_{\text{trees}}$ is well-defined. Note that all the order relations $\leq_{\text{pref}}$, $\leq_{\text{lex}}$, $\leq^2_{\text{lex}}$, $\leq_{\text{llex}}$, and $\leq_{\text{trees}}$ are automatic.

Let $\mathcal{L} = (D,;\leq)$ be a linear order. A nonempty set $I \subseteq D$ is an *interval* if $x, z \in I$ and $x < y < z$ imply $y \in I$ for all $x, y, z \in D$. The linear order $\mathcal{L}$ is *scattered* if there is no embedding of the rational line $(\mathbb{Q};\leq)$ into $\mathcal{L}$. Examples of scattered linear orders are the linear order of the non-negative integers ω, of the non-positive integers ω^*, of all integers ζ, or the linear order of size $n \in \mathbb{N}$ that we denote $\underline{n}$.[1] If Γ is an alphabet with at least 2 letters, then $(\Gamma^*;\leq_{\text{llex}}) \cong \omega$ is scattered, too. On the other hand, if $a, b \in \Gamma$ are distinct letters, then $(\{aa, bb\}^*ab;\leq_{\text{lex}}) \cong (\mathbb{Q};\leq)$. Hence $(\Gamma^*;\leq_{\text{lex}})$ is not scattered. From [22, Prop. 4.10], we know that the set of word-automatic presentations of scattered linear orders is decidable.

A linear order $\mathcal{L} = (L;\leq)$ is *rigid* if it does not admit any non-trivial automorphism, i.e., if the identity mapping $f\colon L \to L\colon x \mapsto x$ is the only automorphism of $\mathcal{L}$. The linear orders ω, ω^*, and $\underline{n}$ for $n \in \mathbb{N}$ are all rigid. On the other hand, $(\mathbb{Q};\leq)$ as well as ζ are not rigid.

Note that automorphisms of tree-automatic linear orders are binary relations on the set of all trees. Hence it makes sense to speak of an *automatic automorphism.* A tree-automatic structure is *automatically rigid* if it does not have any non-trivial automatic automorphism.

Let $\mathcal{I} = (I;\leq)$ be a linear order and, for $i \in I$, let $\mathcal{L}_i = (L_i;\leq_i)$ be a linear order. Then the $\mathcal{I}$-*sum*[2] of these linear orders is defined by

$$\sum_{i\in(I;\leq)} \mathcal{L}_i = \left(\biguplus_{i\in I} L_i; \bigcup_{i\in I} \leq_i \cup \bigcup_{\substack{i,j\in I\\ i<j}} (L_i \times L_j) \right) .$$

For $\sum_{i\in\underline{2}} \mathcal{L}_i$, we simply write $\mathcal{L}_1 + \mathcal{L}_2$. If, for all $i \in I$, $\mathcal{L}_i = \mathcal{L}$, then we write $\mathcal{L} \cdot \mathcal{I}$ for $\sum_{i\in(I;\leq)} \mathcal{L}_i$. Note that $\mathcal{L} \cdot \mathcal{I}$ is obtained by replacing every element of $\mathcal{I}$ by a copy of $\mathcal{L}$. As an example, consider the linear order $\delta = \omega \cdot \omega^*$. This linear order will be used as "delimiter" in our constructions. It is isomorphic to $(\mathbb{N} \times \mathbb{N};\leq_\delta)$ with

$$(i,j) \leq_\delta (k,\ell) :\Leftrightarrow j > \ell \text{ or } j = \ell \text{ and } i \leq k\,.$$

Hence it forms a descending chain of ascending chains. Therefore, it has no minimal and no maximal element, is rigid and scattered. Note that the mapping $(i,j) \mapsto 10^{j+1}1^{i+1}0$ is an isomorphism from $(\mathbb{N} \times \mathbb{N};\leq_\delta)$ onto $(10^+1^+0;\leq_{\text{lex}})$ where we assume $0 < 1$. Hence $\delta \cong (10^+1^+0;\leq_{\text{lex}})$.

[1]In this paper, I prefer this non-standard notation over the standard notation $\mathbf{n}$ since it makes it more convenient to denote the linear order of size $f(n)$ where $f\colon \mathbb{N} \to \mathbb{N}$ is some function.

[2]Shuffle sums mentioned in the introduction are special cases of this construction where $\mathcal{I} = (\mathbb{Q};\leq)$ is the rational line and, for every $q \in \mathbb{Q}$, the set $\{r \in \mathbb{Q} \mid \mathcal{L}_q \cong \mathcal{L}_r\}$ is a dense subset of $(\mathbb{Q},\leq)$.

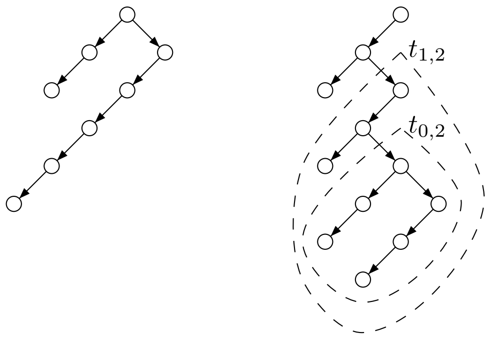

Figure 1: The trees $t_{0,4}$ (at the left) and $t_{2,2}$ (at the right) from D

Also for later use, we next define a regular set $D = \{t_{i,j} \mid i, j \geq 0\}$ of trees such that $\delta \cong (D; \leq_{\text{trees}})$. The alphabet of these trees will be the singleton $\{\$\}$ so that a tree is completely given by its domain. Then set inductively

$$\begin{aligned} \text{dom}(t_{0,j}) &= \{\varepsilon, 0, 00\} \cup 1\{0^k \mid 0 \leq k \leq j\} \text{ and} \\ \text{dom}(t_{i+1,j}) &= \{\varepsilon, 0, 00\} \cup 01\,\text{dom}(t_{i,j}) \end{aligned}$$

The trees $t_{0,4}$ and $t_{2,2}$ are depicted in Figure 1 (left-arrows denote 0-sons, right-arrows denote 1-sons), the trees $t_{1,2}$ and $t_{0,2}$ occur as subtrees of $t_{2,2}$ according to the inductive definition of $t_{2,2}$. The tuple of "side trees" $\tau(t_{i,j})$ has the following form

$$\tau(t_{0,j}) = (t_j, \emptyset, \emptyset) \text{ with } \text{dom}(t_j) = \{0^k \mid 0 \leq k \leq j\} \text{ and } \tau(t_{i+1,j}) = (\emptyset, t_{i,j}, \emptyset)\,.$$

Note that all trees $t_{i,j}$ coincide on their main branch, i.e., $t_{i,j}\restriction_{0^*} = t_{k,\ell}\restriction_{0^*}$. Hence $t_{i,j} \leq_{\text{trees}} t_{k,\ell}$ if and only if $\tau(t_{i,j})$ is lexicographically smaller than $\tau(t_{k,\ell})$. But this is the case if and only if

- $0 = k < i$ or
- $0 = i = k$ and $j \leq \ell$ or
- $0 < i, k$ and $t_{i-1,j} \leq_{\text{trees}} t_{k-1,\ell}$.

By induction, this is equivalent to $i > k$ or $i = k$, $j \leq \ell$. Hence, indeed, $(D; \leq_{\text{trees}}) \cong \delta$.

3. Automorphisms of linear orders on words

In this section, we consider linear orders on sets of words. Namely, we consider regular universes with the linear order $\leq^2_{\text{lex}}$ and contextfree universes ordered lexicographically. In both cases, we prove that the existence of a nontrivial automorphism for scattered linear orders of the respective form is Π^0_1-hard.

3.1. Regular universe and $\leq^2_{\text{lex}}$

Let $p, q \in \mathbb{N}[\bar{x}]$ be two polynomials with coefficients in $\mathbb{N}$ and variables among $\bar{x} = (x_1, \dots, x_k)$. In the following, we identify the polynomial p with its polynomial function $p\colon \mathbb{N}^k \to \mathbb{N}$. Then define the linear order

$$\mathcal{L}_{p,q} = \sum_{\bar{x} \in (\mathbb{N}^k; \leq_{\text{lex}})} \Big((\underline{p(\bar{x})} + \delta) \cdot \omega^* + (\underline{q(\bar{x})} + \delta) \cdot \omega) \Big) \ .$$

The linear order $\mathcal{L}_{p,q}$ is scattered since δ, ω, ω^*, and $(\mathbb{N}^k; \leq_{\text{lex}})$ are all scattered.

Lemma 3.1. *Let $p, q \in \mathbb{N}[\bar{x}]$. Then $\mathcal{L}_{p,q}$ is rigid if and only if $p(\bar{x}) \neq q(\bar{x})$ for all $\bar{x} \in \mathbb{N}^k$.*

Proof. Suppose there is $\bar{x} \in \mathbb{N}^k$ such that $p(\bar{x}) = q(\bar{x}) = n$. Then $\mathcal{L}_{p,q}$ contains an interval of the form

$$(\underline{n} + \delta) \cdot \omega^* + (\underline{n} + \delta) \cdot \omega = (\underline{n} + \delta) \cdot \zeta \ .$$

Since ζ has a nontrivial automorphism, so does this interval and therefore $\mathcal{L}_{p,q}$. This proves the implication "$\Rightarrow$".

Now suppose f is a nontrivial automorphism of $\mathcal{L}_{p,q}$. Let $\mathcal{L}'$ denote the set of maximal finite intervals of $\mathcal{L}_{p,q}$ with the order inherited from $\mathcal{L}_{p,q}$. Since δ has no maximal finite intervals, we get

$$\mathcal{L}' \cong \sum_{\bar{x} \in (\mathbb{N}^k; \leq_{\text{lex}})} (\omega^* + \omega) \cong \zeta \cdot \omega \ .$$

Then f induces an automorphism f' of $\mathcal{L}'$. Recall that $\delta = \omega \cdot \omega^*$ is rigid. Since f is nontrivial, we get that f' is nontrivial. Since $\omega^* + \omega = \zeta$ has no endpoints and since ω is rigid, f' has to map every copy of ζ in $\mathcal{L}'$ onto itself. Since f' is nontrivial, there is $\bar{x} \in \mathbb{N}^k$ such that f' acts nontrivially on the corresponding copy of ζ. Since f' is induced by f, the automorphism f therefore acts nontrivially on

$$(\underline{p(\bar{x})} + \delta) \cdot \omega^* + (\underline{q(\bar{x})} + \delta) \cdot \omega \ .$$

Since f maps maximal finite intervals onto maximal finite intervals, we get $p(\bar{x}) = q(\bar{x})$. □

We now prove that $\mathcal{L}_{p,q}$ is word-automatic or, more specifically, we will construct a regular set $L \subseteq \{0,1\}^+ \otimes \{0,1\}^+$ such that $\mathcal{L}_{p,q} \cong (L; \leq^2_{\text{lex}})$ (see Lemma 3.3 below).

Let $\mathcal{A} = (Q, \iota, \Delta, F)$ be a word-automaton over the alphabet Γ and let $w \in \Gamma^+$ be a word. Then $\text{Run}(\mathcal{A}, w)$ is the set of all words over Δ of the form

$$(q_0, a_1, q_1, \iota)(q_1, a_2, q_2, \iota) \dots (q_{k-1}, a_k, \iota, \iota)$$

with $w = a_1 a_2 \dots a_k$ and $q_0 \in F$. These words encode the accepting runs of the word-automaton $\mathcal{A}$ (recall that word-automata are special bottom up tree-automata which explains the unusual position of the initial and final states in the run). Furthermore, let $\text{Run}(\mathcal{A}) = \bigcup_{w \in \Gamma^+} \text{Run}(\mathcal{A}, w)$.

Lemma 3.2. *From polynomials $p, q \in \mathbb{N}[x_1, \dots, x_k]$, one can construct an ordered alphabet $(\Gamma; \leq)$ and a regular language $K \subseteq \Gamma^+ \otimes \Gamma^+$ such that $(K; \leq^2_{\text{lex}}) \cong \mathcal{L}_{p,q}$.*

If $\mathcal{L}_{p,q}$ has a non-trivial automorphism, $(K; \leq^2_{\text{lex}})$ has a non-trivial automatic automorphism.

Proof. Let p and q be polynomials from $\mathbb{N}[x_1,\ldots,x_k]$. For $\bar{x}=(x_1,\ldots,x_k)\in\mathbb{N}^k$, set

$$a^{\bar{x}} = a^{x_1}¢a^{x_2}¢\cdots¢a^{x_k}¢ \in (a^*¢)^k\,.$$

Then, as in [25, Lemma 7], one can construct word-automata $\mathcal{A}_p=(Q_p,\iota_p,\Delta_p,F_p)$ and $\mathcal{A}_q=(Q_q,\iota_q,\Delta_q,F_q)$ with $L(\mathcal{A}_p),L(\mathcal{A}_q)\subseteq(a^*¢)^k$, such that, for $\bar{x}\in\mathbb{N}^k$, the word-automaton $\mathcal{A}_p$ has precisely $p(\bar{x})$ many accepting runs on the word $a^{\bar{x}}$. In other words, $|\mathrm{Run}(\mathcal{A}_p,a^{\bar{x}})|=p(\bar{x})$ and $|\mathrm{Run}(\mathcal{A}_q,a^{\bar{x}})|=q(\bar{x})$. We will assume $\Delta_p\cap\Delta_q=\emptyset$.

Define the language K by

$$\begin{aligned} K \;=\; & \bigcup_{\bar{x}\in\mathbb{N}^k} a^{\bar{x}}0^+1\otimes(\mathrm{Run}(\mathcal{A}_p,a^{\bar{x}})\cup 32^+3^+2)\\ & \cup\bigcup_{\bar{x}\in\mathbb{N}^k} a^{\bar{x}}1^+0\otimes(\mathrm{Run}(\mathcal{A}_q,a^{\bar{x}})\cup 32^+3^+2)\,.\end{aligned}$$

Hence any word from K is the convolution of two words over the alphabet

$$\Gamma=\{a,¢,0,1,2,3\}\cup\Delta_p\cup\Delta_q\,.$$

We have to show that the language K is effectively regular. We have

$$\begin{aligned} K = \; & \bigcup_{\bar{x}\in\mathbb{N}^k}\left(a^{\bar{x}}0^+1\otimes\mathrm{Run}(\mathcal{A}_p,a^{\bar{x}})\right)\cup\left(\bigcup_{\bar{x}\in\mathbb{N}^k}a^{\bar{x}}0^+1\right)\otimes 32^+3^+2\\ & \cup\bigcup_{\bar{x}\in\mathbb{N}^k}\left(a^{\bar{x}}1^+0\otimes\mathrm{Run}(\mathcal{A}_q,a^{\bar{x}})\right)\cup\left(\bigcup_{\bar{x}\in\mathbb{N}^k}a^{\bar{x}}1^+0\right)\otimes 23^+2^+3\end{aligned}$$

Note that the convolution of the direct product $L_1\times L_2$ of two regular languages is always regular. Hence, the crucial point here is the regularity of the set

$$\bigcup_{\bar{x}\in\mathbb{N}^k} a^{\bar{x}}0^+1\otimes\mathrm{Run}(\mathcal{A}_p,a^{\bar{x}})\,.$$

Note that for any words $w=a^{\bar{x}}$ and $W\in\mathrm{Run}(\mathcal{A}_p,w)$, we have $|w|=|W|$. Therefore, the set in question equals

$$\left[\bigcup_{\bar{x}\in\mathbb{N}^k} a^{\bar{x}}\otimes\mathrm{Run}(\mathcal{A}_p,a^{\bar{x}})\right]\cdot\left(0^+1\otimes\{\varepsilon\}\right)\,.$$

But a word belongs to the language in square brackets if and only if it is the convolution of a word w from the regular language $(a^*¢)^k$ and a run of the automaton $\mathcal{A}_p$ on this word w, a property that a word-automaton can check easily.

On the alphabet Γ, we now fix a linear order $\leq$ such that

$$\Delta_p\cup\Delta_q<0<1<2<3<¢<a\,.$$

The associated order $\leq^2_{\mathrm{lex}}$ on the language K can now be characterized as follows:

$$a^{\bar{x}}b^m(1-b)\otimes r\leq^2_{\mathrm{lex}} a^{\bar{y}}c^n(1-c)\otimes s$$

(with $b,c\in\{0,1\}$, $\bar{x},\bar{y}\in\mathbb{N}^k$, and $m,n>0$) if and only if

(i) $b = 0$, $c = 1$ and $\bar{x} \leq_{\text{lex}} \bar{y}$, or

(ii) $b = 1$, $c = 0$ and $\bar{x} <_{\text{lex}} \bar{y}$, or

(iii) $b = c$ and

(iii.1) $\bar{x} <_{\text{lex}} \bar{y}$, or

(iii.2) $\bar{x} = \bar{y}$, $b = 0$, and $m > n$, or

(iii.3) $\bar{x} = \bar{y}$, $b = 1$, and $m < n$, or

(iii.4) $\bar{x} = \bar{y}$, $m = n$, and

(iii.4.1) $r \in \text{Run}(\mathcal{A}_p) \cup \text{Run}(\mathcal{A}_q)$ and $s \in 32^+3^+2$, or

(iii.4.2) $r, s \in \text{Run}(\mathcal{A}_p) \cup \text{Run}(\mathcal{A}_q)$ and $r \leq_{\text{lex}} s$, or

(iii.4.3) $r, s \in 32^+3^+2$ and $r \leq_{\text{lex}} s$.

We show $(K; \leq^2_{\text{lex}}) \cong \mathcal{L}_{p,q}$. For $\bar{x} \in \mathbb{N}^k$ and $m \geq 1$, let $\mathcal{I}_{\bar{x},0^m1}$ denote the restriction of $(K; \leq^2_{\text{lex}})$ to the set $a^{\bar{x}}0^m1 \otimes (\text{Run}(\mathcal{A}_p, a^{\bar{x}}) \cup 32^+3^+2) \subseteq K$. By (iii.4.1), $\mathcal{I}_{\bar{x},0^m1}$ is isomorphic to the sum of the restrictions of $(K; \leq^2_{\text{lex}})$ to the sets $a^{\bar{x}}0^m1 \otimes \text{Run}(\mathcal{A}_p, a^{\bar{x}})$ and $a^{\bar{x}}0^m1 \otimes 32^+3^+2$, respectively. By (iii.4.2) and the choice of the automaton $\mathcal{A}_p$, the first restriction is isomorphic to $\underline{p(\bar{x})}$. Recall that $(32^+3^+2; \leq_{\text{lex}}) \cong \delta$. Hence, the second restriction is isomorphic to δ by (iii.4.3). In summary,

$$\mathcal{I}_{\bar{x},0^m1} \cong \underline{p(\bar{x})} + \delta\,.$$

Next, for $\bar{x} \in \mathbb{N}^k$, let $\mathcal{I}_{\bar{x},0^+1}$ denote the restriction of $(K; \leq^2_{\text{lex}})$ to the set $a^{\bar{x}}0^+1 \otimes (\text{Run}(\mathcal{A}_p, a^{\bar{x}}) \cup 32^+3^+2)$. Note that, by (iii.2), we have

$$\mathcal{I}_{\bar{x},0^{m+1}1} <^2_{\text{lex}} \mathcal{I}_{\bar{x},0^m1}$$

for all $m \geq 1$. Hence

$$\mathcal{I}_{\bar{x},0^+1} = \sum_{m \leq -1} \mathcal{I}_{\bar{x},0^{-m}1} \cong (\underline{p(\bar{x})} + \delta) \cdot \omega^*\,.$$

With $\mathcal{I}_{\bar{x},1^+0}$ the restriction of $(K; \leq^2_{\text{lex}})$ to the set $a^{\bar{x}}1^+0 \otimes (\text{Run}(\mathcal{A}_q, a^{\bar{x}}) \cup 32^+3^+2)$, we obtain similarly

$$\mathcal{I}_{\bar{x},1^+0} = \sum_{m \geq 1} \mathcal{I}_{\bar{x},1^m0} \cong (\underline{q(\bar{x})} + \delta) \cdot \omega$$

(the reason for the factor ω instead of ω^* above is the difference between (iii.2) and (iii.3)).

Finally, for $\bar{x} \in \mathbb{N}^k$, let $\mathcal{I}_{\bar{x}}$ denote the restriction of $(K; \leq^2_{\text{lex}})$ to the set

$$\left[a^{\bar{x}}0^+1 \otimes (\text{Run}(\mathcal{A}_p, a^{\bar{x}}) \cup 32^+3^+2)\right] \cup \left[a^{\bar{x}}1^+0 \otimes (\text{Run}(\mathcal{A}_q, a^{\bar{x}}) \cup 32^+3^+2)\right].$$

Then (i) with $\bar{x} = \bar{y}$ and the above imply

$$\mathcal{I}_{\bar{x}} = \mathcal{I}_{\bar{x},0^+1} + \mathcal{I}_{\bar{x},1^+0} \cong (\underline{p(\bar{x})} + \delta) \cdot \omega^* + (\underline{q(\bar{x})} + \delta) \cdot \omega\,.$$

Together with (i), (ii), and (iii.1), this ensures

$$(K; \leq_{\text{lex}}^2) = \sum_{\bar{x} \in (\mathbb{N}^k; \leq_{\text{lex}})} \mathcal{I}_{\bar{x}} \cong \mathcal{L}_{p,q} \,.$$

This finishes the proof of the first claim.

Now suppose that $\mathcal{L}_{p,q}$ has a non-trivial automorphism. By Lemma 3.1, there is $\bar{y} \in \mathbb{N}^k$ such that $p(\bar{y}) = q(\bar{y})$. From the construction of the automata $\mathcal{A}_p$ and $\mathcal{A}_q$, we infer $|\text{Run}(\mathcal{A}_p, a^{\bar{y}})| = |\text{Run}(\mathcal{A}_q, a^{\bar{y}})|$. Let

$$\text{Run}(\mathcal{A}_p, a^{\bar{y}}) = \{\rho_1, \ldots, \rho_n\} \text{ and } \text{Run}(\mathcal{A}_q, a^{\bar{y}}) = \{\sigma_1, \ldots, \sigma_n\}$$

with

$$\rho_1 <_{\text{lex}} \rho_2 <_{\text{lex}} \cdots <_{\text{lex}} \rho_n \text{ and } \sigma_1 <_{\text{lex}} \sigma_2 <_{\text{lex}} \cdots <_{\text{lex}} \sigma_n \,.$$

Now define a mapping $f \colon K \to K$ by

$$f(a^{\bar{x}} b^m (1-b) \otimes r) = \begin{cases} a^{\bar{x}} b^m (1-b) \otimes r & \text{if } \bar{x} \neq \bar{y} \\ a^{\bar{y}} b^{m-1} (1-b) \otimes r & \text{if } \bar{x} = \bar{y}, b = 0, m > 1 \\ a^{\bar{y}} 10 \otimes r & \text{if } \bar{x} = \bar{y}, b = 0, m = 1, r \in 32^+3^+2 \\ a^{\bar{y}} 10 \otimes \sigma_i & \text{if } \bar{x} = \bar{y}, b = 0, m = 1, r = \rho_i \\ a^{\bar{y}} b^{m+1} (1-b) \otimes r & \text{if } \bar{x} = \bar{y}, b = 1 \end{cases}$$

This mapping fixes all elements of K not belonging to $\mathcal{I}_{\bar{y}}$. On this linear order $\mathcal{I}_{\bar{y}}$, it acts as an increasing automorphism. Hence f is a non-trivial automorphism of $(K; \leq_{\text{llex}}^2)$. It is not hard to verify that $f^{\otimes}$ is regular. □

Lemma 3.3. *From polynomials $p, q \in \mathbb{N}[x_1, \ldots, x_k]$, one can construct a regular language $L \subseteq \{0,1\}^+ \otimes \{0,1\}^+$ such that $(L; \leq_{\text{lex}}^2) \cong \mathcal{L}_{p,q}$.*

If $\mathcal{L}_{p,q}$ has a non-trivial automorphism, then it has a non-trivial automatic automorphism.

Proof. Let $p, q \in \mathbb{N}[x_1, \ldots, x_k]$ be polynomials, let K and $(\Gamma; \leq)$ be the language and the ordered alphabet from Lemma 3.2. Furthermore, let $(\Gamma; \leq)$ be the sequence

$$\sigma_1 < \sigma_2 < \cdots < \sigma_\ell \,.$$

Let g denote the monoid homomorphism from Γ^* to $\{0,1\}^*$ defined by $g(\sigma_i) = 1^i 0^{\ell - i}$ for $1 \leq i \leq \ell$. Now set $L = \{g(u) \otimes g(v) \mid u \otimes v \in K\}$. Then g is an isomorphism from $(K; \leq_{\text{lex}}^2)$ onto $(L; \leq_{\text{lex}}^2)$. Since all the words $g(\sigma_i)$ have the same length, the language L is also regular.

If $\mathcal{L}_{p,q}$ has a non-trivial automorphism, then, by Lemma 3.2, there is a non-trivial automorphism f of $(K; \leq_{\text{llex}}^2)$ such that $f^{\otimes}$ is regular. Hence $g = h^{-1} \circ f \circ h$ is a non-trivial automorphism of $(L; \leq_{\text{llex}}^2)$. Note that $h^{\otimes}$ is regular. It follows that also $g^{\otimes}$ is regular. □

Theorem 3.4. *(i) The set of regular languages $L \subseteq \{0,1\}^+ \otimes \{0,1\}^+$ such that $(L; \leq_{\text{lex}}^2)$ is rigid (is rigid and scattered, respectively), is Π_1^0-hard.*

(ii) *The set of regular languages $L \subseteq \{0,1\}^+ \otimes \{0,1\}^+$ such that $(L; \leq^2_{\text{lex}})$ is automatically rigid (automatically rigid and scattered, respectively) is Π^0_1-hard.*

PROOF. (i) The set of pairs of polynomials $p, q \in \mathbb{N}[\bar{x}]$ with $p(\bar{y}) \neq q(\bar{y})$ for all $\bar{y} \in \mathbb{N}^k$ is Π^0_1-complete [28]. We reduce this to the first set in question: Let $p, q \in \mathbb{N}[\bar{x}]$ and let L be the regular language from Lemma 3.3. Then $(L; \leq^2_{\text{lex}}) \cong \mathcal{L}_{p,q}$ is rigid if and only if $p(\bar{y}) \neq q(\bar{y})$ for all $\bar{y} \in \mathbb{N}^k$ by Lemma 3.1.

Note that this is even a reduction to the second set in question since the linear order $\mathcal{L}_{p,q}$ is scattered.

(ii) By Lemma 3.3, $\mathcal{L}_{p,q}$ is rigid if and only if $(L; \leq^2_{\text{llex}})$ is automatically rigid. Hence the above reduction also proves the two claims from (ii). □

Corollary 3.5. *(i) The set of word-automatic presentations of rigid (rigid and scattered, respectively) linear orders is Π^0_1-hard.*

(ii) The set of word-automatic presentations of automatically rigid (automatically rigid and scattered, respectively) linear orders is Π^0_1-complete.

PROOF. The two claims from (i) are obvious consequences of Theorem 3.4(i). Analogously, the two hardness claims from (ii) follow immediately from Theorem 3.4(ii).

Now let $(L; \leq)$ be a word-automatic linear order given by a word-automatic presentation over the alphabet Γ. Let $R \subseteq \Gamma^+ \times \Gamma^+$. Then it can be expressed in first-order logic that R is a non-trivial automorphism of $(L; \leq)$. Hence, given a word-automaton $\mathcal{A}$ for a regular language $R^{\otimes} \subseteq \Gamma^+ \otimes \Gamma^+$, one can decide whether R is a non-trivial automorphism of $(L; \leq)$ [19]. Consequently, automatic rigidity of $(L; \leq)$ is a Π^0_1-property.

□

3.2. Contextfree universe and $\leq_{\text{lex}}$

Ésik initiated the investigation of linear orders of the form $(L; \leq_{\text{lex}})$ where L is contextfree. Density of such a linear order is undecidable [13], the isomorphism problem is Σ^1_1-complete [25], their rank is bounded by ω^ω [7], and there is a contextfree language L such that the first-order theory of $(L; \leq_{\text{lex}})$ is undecidable [6]. Even more, there exist one-counter languages L_1 and L_2 such that the Σ_3-theory of $(L_1; \leq_{\text{lex}})$ is undecidable and the first-order theory of $(L_2; \leq_{\text{lex}})$ is non-arithmetical [24].

We will show that rigidity of $(L; \leq_{\text{lex}})$ is undecidable for contextfree languages L. The proof uses the linear order $\mathcal{L}_{p,q}$ and constructs a deterministic contextfree language L' such that $(L'; \leq_{\text{lex}}) \cong \mathcal{L}_{p,q}$. This construction is a variant of the construction in the proof of Lemma 3.2.

Lemma 3.6. *From polynomials $p, q \in \mathbb{N}[x_1, \ldots, x_k]$, one can construct a deterministic contextfree language $L' \subseteq \{0,1\}^+$ such that $(L'; \leq_{\text{lex}}) \cong \mathcal{L}_{p,q}$.*

PROOF. Let $p, q \in \mathbb{N}[x_1, \ldots, x_k]$ be polynomials and let K and $(\Gamma; \leq)$ be the language and the ordered alphabet from Lemma 3.2. Then set

$$K' = \{u\$v^{rev} \mid u \otimes v \in K\}$$

where v^{rev} is the reversal of the word v. Then, from a deterministic finite automaton $\mathcal{A}$ accepting K^{rev}, one can construct a deterministic pushdown automaton accepting K' (reading $u\$v$, it stores u in the stack and, after reading \$, simulates $\mathcal{A}$ while emptying the stack). Note that the alphabet of K' is

$$\Gamma' = \{\$\} \cup \Gamma = \{\$, a, ¢, 0, 1, 2, 3\} \cup \Delta_p \cup \Delta_q \,.$$

We order the alphabet Γ' by $\leq'$ such that

$$\Delta_p \cup \Delta_q <' 0 <' 1 <' 3 <' 2 <' ¢ <' a <' \$\,.$$

Compared to the proof of Lemma 3.2, the order of 2 and 3 is inverted and \$ is made the new maximal element (we could have placed \$ anywhere). With $\leq$ the order on Γ from the proof of Lemma 3.2, one effect of this definition is

$$((32^+3^+2)^{rev}; \leq'_{\text{lex}}) \cong (32^+3^+2; \leq_{\text{lex}}) \cong \delta$$

which will be used in the third item below.

To show $(K'; \leq'_{\text{lex}}) \cong \mathcal{L}_{p,q}$, is suffices to prove $(K'; \leq'_{\text{lex}}) \cong (K; \leq^2_{\text{lex}})$. For this, recall that $(K; \leq^2_{\text{lex}})$ is a sequence of the following blocks (for $\bar{x} \in \mathbb{N}^k$ and $m \geq 1$):

- $(a^{\bar{x}}0^m1 \otimes \text{Run}(\mathcal{A}_p, a^{\bar{x}}); \leq^2_{\text{lex}})$: This linear order is finite of size $|\text{Run}(\mathcal{A}_p, a^{\bar{x}})|$. The same holds of the linear order

$$(a^{\bar{x}}0^m1\$\{r^{rev} \mid r \in \text{Run}(\mathcal{A}_p, a^{\bar{x}})\}; \leq'_{\text{lex}})\,.$$

- $(a^{\bar{x}}1^m0 \otimes \text{Run}(\mathcal{A}_q, a^{\bar{x}}); \leq^2_{\text{lex}})$: As above, this is isomorphic to

$$(a^{\bar{x}}1^m0\$\{r^{rev} \mid r \in \text{Run}(\mathcal{A}_q, a^{\bar{x}})\}; \leq'_{\text{lex}})\,.$$

- $(a^{\bar{x}}b^m(1-b) \otimes 32^+3^+2; \leq^2_{\text{lex}})$ (for $b \in \{0,1\}$) which is isomorphic to δ. But δ is also isomorphic to

$$(a^{\bar{x}}b^m(1-b)\$23^+2^+3; \leq'_{\text{lex}})\,.$$

It therefore follows that $(K; \leq^2_{\text{lex}})$ and $(K'; \leq'_{\text{lex}})$ are isomorphic. The construction of $L' \subseteq \{0,1\}^+$ then follows the proof of Lemma 3.3. □

Now we obtain, in the same way that we proved Theorem 3.4(i), the following result.

Theorem 3.7. *The set of deterministic contextfree languages $L \subseteq \{0,1\}^+$ such that $(L; \leq_{\text{lex}})$ is rigid (is rigid and scattered, respectively), is Π^0_1-hard.*

4. Isomorphisms and automorphisms of linear orders on trees

In this section, we will show that the isomorphism of scattered and tree-automatic linear orders is undecidable. Furthermore, we will prove that the existence of a non-trivial automorphism in this case is Σ^0_2-hard. Both these results use (an improved version of) a theorem by Krob [23] that we discuss first. Our discussion elaborates ideas from [1] where Krob's theorem is shown, but not our strengthenings Theorem 4.8 and Corollary 4.7.

4.1. Weighted automata and two-counter machines

4.1.1. Definitions and examples

A *weighted automaton* is a tuple $\mathcal{A} = (Q, \Gamma, \iota, \mu, F)$ where Q is the finite set of states, Γ the alphabet, $\iota \in Q$ is the initial state, $F \subseteq Q$ is the set of accepting states, and $\mu \colon Q \times \Gamma \times Q \to \mathbb{Z} \cup \{-\infty\}$ is the weight function. If $H \subseteq \mathbb{Z}$ and the image of μ is contained in $H \cup \{-\infty\}$, then we speak of an H-*weighted automaton*.

A *run* of $\mathcal{A}$ is a sequence $\rho = (q_0, a_1, q_1) \dots (q_{k-1}, a_k, q_k) \in \Delta^+$ with $q_0 = \iota$, $q_k \in F$, and $\mu(q_{i-1}, a_i, q_i) \neq -\infty$ for all $1 \leq i \leq k$. Its *label* is the word $a_1 \dots a_k \in \Gamma^+$. By $\mathrm{Run}(\mathcal{A}, w)$ we denote the set of runs labeled w and $\mathrm{Run}(\mathcal{A})$ denotes the set of all runs of $\mathcal{A}$. The *weight* $\mathrm{wt}(\rho)$ of the run ρ is the sum of the weights of the transitions, i.e.,

$$\mathrm{wt}(\rho) = \sum_{1 \leq i \leq k} \mu(q_{i-1}, a_i, q_i) \in \mathbb{Z}\,.$$

The *behaviour* $||\mathcal{A}||$ of $\mathcal{A}$ is the function from Γ^+ to $\mathbb{N} \cup \{-\infty\}$ that maps the word w to the maximal weight of a run with label w and to $-\infty$ if no such run exists.

Notation. In this section, we consider the following alphabets:

$$\begin{aligned}\Gamma_0 &= \{\mathbf{+}_1, \mathbf{+}_2, \mathbf{-}_1, \mathbf{-}_2, \mathbf{0}_1, \mathbf{0}_2\}\\ \Gamma_1 &= \Gamma_0 \cup \{\#\}\\ \Gamma &= \Gamma_1 \cup \{\Box\}\end{aligned}$$

Furthermore, $|u|_a$ for $u \in \Gamma^*$ and $a \in \Gamma$ denotes the number of occurrences of the letter a in the word u.

Example 4.1. *Consider the $\{-1, 0, 1\}$-weighted automaton $\mathcal{A}^1$ from Fig. 2.*[3] *Let $w \in \Gamma^*$ be a word and consider any w-labeled run from the initial state ι to any of the final states ι and f. Let u be the prefix of w that is read until the run leaves ι and let $w = uv$. Then the weight of the run equals*

$$|u|_{\mathbf{-}_1} - |u|_{\mathbf{+}_1}\,.$$

Since the automaton can leave the state ι towards state f at any time, we get

$$||\mathcal{A}_1||(w) = \max\{|u|_{\mathbf{-}_1} - |u|_{\mathbf{+}_1} \mid u \leq_{\mathrm{pref}} w\}\,.$$

In the proof of Lemma 4.4, we will use that $||\mathcal{A}^1||(w) > 0$ if and only if there exists a prefix u of w with $|u|_{\mathbf{-}_1} - |u|_{\mathbf{+}_1} > 0$, i.e., with $|u|_{\mathbf{-}_1} > |u|_{\mathbf{+}_1}$.

Example 4.2. *Let $\mathcal{B}_1^1$ be the first $\{-1, 0, 1\}$-weighted automaton of Fig. 3 and let $\mathcal{B}_2^1$ be the second one. Note that $\mathcal{B}_1^1$ differs from $\mathcal{A}^1$ in two aspects, only: the initial state ι is not final anymore and the label of the transition from ι to f changed from Γ_1 to $\mathbf{0}_1$. Then, similarly to the arguments in Example 4.1, we get*

$$||\mathcal{B}_1^1||(w) = \max\{|u|_{\mathbf{-}_1} - |u|_{\mathbf{+}_1} \mid u\mathbf{0}_1 \leq_{\mathrm{pref}} w\}\,.$$

[3] In this figure, the annotation $a/1$ at a transition denotes that it is labeled by the letter a and carries weight 1, $X/1$ for $X \subseteq \Gamma$ denotes that there are transitions for all the letters from X with weight 1.

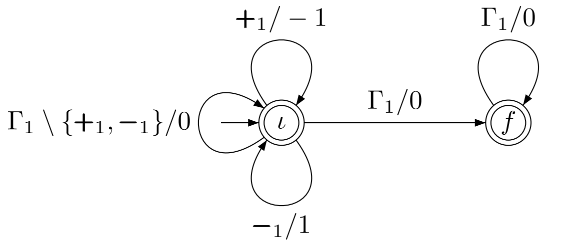

Figure 2: The automaton $\mathcal{A}^1$ from Example 4.1

The difference between $\mathcal{B}_1^1$ and $\mathcal{B}_2^1$ is the weight of the loops labeled $+_1$ and $-_1$ at the state ι: these weights are exchanged. Hence we obtain

$$||\mathcal{B}_2^1||(w) = \max\{|u|_{+_1} - |u|_{-_1} \mid u\mathbf{0}_1 \leq_{\text{pref}} w\}\,.$$

Now let $\mathcal{B}^1$ be the disjoint union of $\mathcal{B}_1^1$ and $\mathcal{B}_2^1$, i.e., the whole of the automaton in Fig. 3. Then, for $w \in \Gamma_1^*$, we have $||\mathcal{B}^1||(w) \leq 0$ if and only if

$$|u|_{-_1} \leq |u|_{+_1} \leq |u|_{-_1}$$

and therefore

$$|u|_{+_1} = |u|_{-_1}$$

for any prefix $u\mathbf{0}_1$ of w. In the proof of Lemma 4.4, we will use that $||\mathcal{B}^1||(w) > 0$ if and only if there exists a prefix $u\mathbf{0}_1$ of w with $|u|_{-_1} \neq |u|_{+_1}$.

4.1.2. Two-counter machines

A *two-counter machine* is a tuple $M = (I_1, I_2, \dots, I_m)$ where every I_j is of one of the following forms:

- halt
- $z := z + 1;$ goto ℓ
- if $z = 0$ then goto k else $z := z - 1;$ goto ℓ endif

where $z \in \{x_1, x_2\}$ and $1 \leq k, \ell \leq m$. The instruction I_i is considered as the i^{th} line of the program M. These instructions use two counters x_1 and x_2.

A word $w = a_1 a_2 \dots a_n \in \Gamma^*$ *conforms to the control flow of M* if there are "line numbers" $p_0, p_1, \dots, p_n \in \{1, 2, \dots, m\}$ with $p_0 = 1$ such that, for all lines $1 \leq k < n$ and all counters $c \in \{1, 2\}$, the following hold:

- If $a_i = +_c$, then $I_{p_{i-1}} = (x_c := x_c + 1;$ goto $p_i)$.
- If $a_i = -_c$, then $I_{p_{i-1}}$ is of the form if $x = 0$ then goto k else $x_c := x_c - 1;$ goto p_i endif for some $1 \leq k \leq m$.

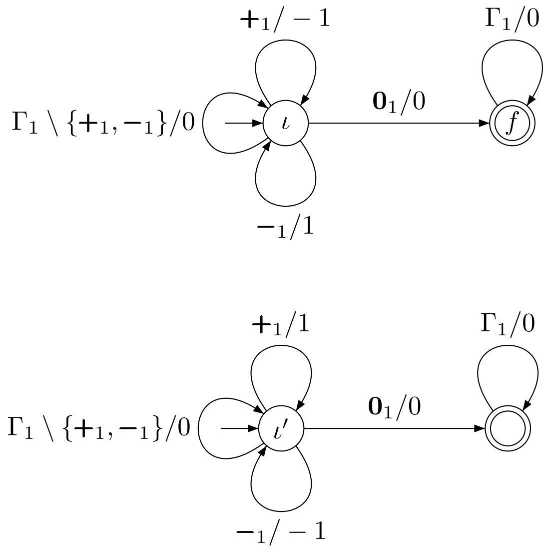

Figure 3: The automaton $\mathcal{B}^1$ from Example 4.2

- If $a_i = \mathbf{0}_c$, then $I_{p_{i-1}}$ is of the form if $x_c = 0$ then goto p_i else $x_c := x_c - 1$; goto k endif for some $1 \leq k \leq m$.
- If $a_i \in \{\square, \#\}$, then $p_{i-1} = p_i$.

Furthermore, $I_{p_n} =$ halt. Intuitively, the word w describes the sequence of atomic actions performed by the program: $+_c$ stands for "counter x_c is incremented, $-_c$ for "counter x_c is decremented, $\mathbf{0}_c$ for "counter x_c is empty", and $\square$ and # for "noop" or "skip".

For $m \in \mathbb{N}$, the word $w \in \Gamma^*$ *conforms to the counter conditions from* m if

- $|u|_{-_1} \leq m + |u|_{+_1}$ and $|u|_{-_2} \leq |u|_{+_2}$ for all prefixes u of w (i.e., the first counter is initialized with m and none of the counters ever carries a negative value),
- $|u|_{-_1} = m + |u|_{+_1}$ for all prefixes $u\mathbf{0}_1$ of w (i.e, the first counter is 0 whenever w claims a successful test for its emptiness), and
- $|u|_{-_2} = |u|_{+_2}$ for all prefixes $u\mathbf{0}_2$ of w.

Finally, the word $w \in \Gamma^*$ is a *halting computation of* M *from* m if it conforms to the control flow of M and to the counter conditions from m. A number $m \in \mathbb{N}$ is *accepted by* M or *belongs to the halting set of* M if there exists a halting computation of M from m.

The crucial property of two-counter machines was shown by Minsky: from a Turing machine, one can construct a two-counter machine that halts on input $2^n \cdot m \in \mathbb{N}$ where m is odd if and only if the Turing machine accepts n. Hence we get

Theorem 4.3 (Minsky [30]). *(i) There exists a two-counter machine M whose halting set is Σ^0_1-complete.*

(ii) The set of two-counter machines M that halt on every input is Π_2^0-complete.

4.1.3. From two-counter machines to weighted automata

Lemma 4.4. *From a two-counter machine M, one can construct a $\{-1, 0, 1\}$-weighted automaton $\mathcal{C}_M$ over Γ_1 such that the following holds for all $w = \mathbf{+}_1{}^m\#u$ with $m \in \mathbb{N}$ and $u \in \Gamma_0^*$:*

$$||\mathcal{C}_M||(w) > 0 \iff u \text{ is no halting computation from } m$$

Note that we do not care about the behavior of $\mathcal{C}_M$ at words w not from $\mathbf{+}_1{}^*\#\Gamma_0^*$.

Proof. Let $M = (I_1, I_2, \dots, I_m)$ be a two-counter machine.

Note that the set of words from Γ_0 that conform to the control flow of M is a regular language. Therefore, also the set L of words $\mathbf{+}_1{}^m\#u$ with $m \in \mathbb{N}$ and $u \in \Gamma_0^*$ such that u does not conform to the control flow of M is regular. Let $\mathcal{A}'$ be some deterministic finite automaton accepting this set. Then, weighting all transitions of $\mathcal{A}$ by 1, we obtain a weighted automaton $\mathcal{A}$ such that (for all $w \in \Gamma_1^+$)

$$||\mathcal{A}||(w) = \begin{cases} |w| & \text{if } w \in L \\ -\infty & \text{otherwise} \end{cases}$$

and therefore

$$||\mathcal{A}||(w) > 0 \iff w \in L\,.$$

Now let $\mathcal{C}_M$ denote the disjoint union of the $\{-1, 0, 1\}$-weighted automata $\mathcal{A}$, $\mathcal{A}^1$ from Example 4.1, $\mathcal{B}^1$ from Example 4.2, as well as $\mathcal{A}^2$ and $\mathcal{B}^2$ that are similar to $\mathcal{A}^1$ and $\mathcal{B}^1$, but replace $\mathbf{+}_1$ by $\mathbf{+}_2$, $\mathbf{-}_1$ by $\mathbf{-}_2$, and $\mathbf{0}_1$ by $\mathbf{0}_2$. Then, for any word $w = \mathbf{+}_1{}^m\#u$ with $m \in \mathbb{N}$ and $u \in \Gamma_0^*$, we have $||\mathcal{C}_M||(w) > 0$ if and only if

$$\max(||\mathcal{A}||(w), ||\mathcal{A}^1||(w), ||\mathcal{A}^2||(w), ||\mathcal{B}^1||(w), ||\mathcal{B}^2||(w)) > 0\,.$$

But this is the case if and only if

(1) u does not conform to the control flow of M (iff $||\mathcal{A}||(w) > 0$), or

(2) w does not conform to the counter conditions from 0 (iff $||\mathcal{A}^1||(w) > 0$, $||\mathcal{A}^2||(w) > 0$, $||\mathcal{A}^2||(w) > 0$, or $||\mathcal{A}^2||(w) > 0$).

Note that item (2) is equivalent to saying that u does not conform to the counter conditions from m. Consequently, (1) and (2), and therefore $||\mathcal{C}_M||(w) > 0$, is equivalent to saying "u is not a halting computation from m". □

A new strengthening of Krob's result is the following:

Theorem 4.5. *There are $\{-1, 0, 1\}$-weighted automata $\mathcal{C}_1$ and $\mathcal{C}_2$ over Γ such that the set of natural numbers m with*

$$||\mathcal{C}_1||(\mathbf{+}_1{}^m\#u) = ||\mathcal{C}_2||(\mathbf{+}_1{}^m\#u) \tag{1}$$

for all $u \in \Gamma_0^*$ is undecidable.

PROOF. By Theorem 4.3, there exists a two-counter machine M with undecidable halting set. Let $\mathcal{C}_1 = \mathcal{C}_M$ be the $\{-1,0,1\}$-weighted automaton from Lemma 4.4. Furthermore, let $\mathcal{C}$ be a $\{-1,0,1\}$-weighted automaton that maps every nonempty word to 1 and ε to $-\infty$ (e.g., $\mathcal{C}$ has an initial state ι, a final state f, a 1-weighted transition for any letter from ι to f, and a 0-weighted loop at f for every letter). Finally, let $\mathcal{C}_2$ be the disjoint union of $\mathcal{C}_M$ and $\mathcal{C}$.

For $u \in \Gamma_0^*$, we have

$$\begin{aligned}||\mathcal{C}_2||(+_1{}^m\#u) &= \max(||\mathcal{C}_M||(+_1{}^m\#u), ||\mathcal{C}||(+_1{}^m\#u)) \\ &= \max(||\mathcal{C}_M||(+_1{}^m\#u), 1)\end{aligned}$$

and therefore

$$\begin{aligned}||\mathcal{C}_1||(+_1{}^m\#u) = ||\mathcal{C}_2||(+_1{}^m\#u) \\ &\iff ||\mathcal{C}_M||(+_1{}^m\#u) > 0 \\ &\iff u \text{ is no halting computation from } m \text{ (by Lemma 4.4)}\,.\end{aligned}$$

Consequently, (1) holds for all $u \in \Gamma_0^*$ if and only if m is not in the halting set of M. $\square$

Since $+_1{}^m\#\Gamma_0^*$ is regular, it is therefore undecidable whether $||\mathcal{C}_1||$ and $||\mathcal{C}_2||$ agree on a given regular language. We next want to prove this statement for $\{0,1\}$-weighted automata. The core of the proof is the following lemma.

Lemma 4.6. *From a $\{-1,0,1\}$-weighted automaton $\mathcal{C}$ over Γ_1, one can construct a $\{0,1\}$-weighted automaton $\mathcal{D}$ over $\Gamma = \Gamma_1 \cup \{\square\}$ such that*

$$||\mathcal{D}||(w) = \begin{cases} ||\mathcal{C}||(a_1 \dots a_k) + k & \text{if } w = a_1\square\, a_2\square\, \dots\, a_k\square \in (\Gamma_1\,\square)^+ \\ & \text{ and } ||\mathcal{C}||(a_1 \dots a_k) > -\infty \\ -\infty & \text{otherwise}\,. \end{cases}$$

PROOF. In a first step, add 1 to every transition – this results in a $\{0,1,2\}$-weighted automaton that assigns $||\mathcal{C}||(w) + |w|$ to every nonempty word. In a second step, split every a-transition into an a-transition and a subsequent $\square$-transition such that the sum of the weights of these two transitions equals the weight of the original transition. More formally, suppose $\mathcal{C} = (Q, \Gamma_1, \iota, \mu, F)$. Then set $Q' = Q \cup (Q \times \Gamma_1 \times Q)$ and define (for $a \in \Gamma_1$ and $p, q \in Q$)

$$\mu'(p, a, (p,a,q)) = \begin{cases} 0 & \text{if } \mu(p,a,q) = -1 \\ 1 & \text{if } \mu(p,a,q) > -1 \\ -\infty & \text{otherwise} \end{cases}$$

$$\mu'((p,a,q), \square, q) = \begin{cases} 0 & \text{if } \mu(p,a,q) < 1 \\ 1 & \text{if } \mu(p,a,q) = 1 \\ -\infty & \text{otherwise} \end{cases}$$

(all other transition weights are $-\infty$). Then $\mathcal{D} = (Q', \Gamma, \iota, \mu', F)$ has the desired properties. $\square$

If $L \subseteq \Gamma_1$ is regular, so is the set of all words $a_1\Box\, a_2\Box\, \ldots\, a_n\Box$ with $a_1a_2\ldots a_n \in L$. Hence, from Theorem 4.5 and Lemma 4.6, we get immediately the following result.

Corollary 4.7. *There are $\{0,1\}$-weighted automata $\mathcal{D}_1$ and $\mathcal{D}_2$ over Γ such that the set of regular languages $L \subseteq \Gamma^*$ with $||\mathcal{D}_1||(w) = ||\mathcal{D}_2||(w)$ for all $w \in L$ is undecidable.*

Using standard and simple techniques for weighted automata, we can now prove the strengthening of Krob's theorem from [9] that we will use later for the isomorphism problem.

Theorem 4.8. *There is a $\{0,1\}$-weighted automaton $\mathcal{D}$ such that the set of $\{0,1\}$-weighted automata $\mathcal{E}$ with $||\mathcal{D}|| = ||\mathcal{E}||$ is Π^0_1-complete.*

PROOF. Let $\mathcal{D}$ be the weighted automaton $\mathcal{D}_1$ from Corollary 4.7. Now let $L \subseteq \Gamma^*$ be regular. We define a new function $f\colon \Gamma^* \to \mathbb{N} \cup \{-\infty\}$ by

$$f(w) = \begin{cases} ||\mathcal{D}||(w) & \text{if } w \notin L \\ ||\mathcal{D}_2||(w) & \text{if } w \in L\,. \end{cases}$$

From a deterministic finite automaton accepting L, one can construct a $\{0,1\}$-weighted automaton $\mathcal{E}_L$ with $||\mathcal{E}_L|| = f$ (cf. [34, Theorem 4.13]). Then $||\mathcal{D}|| = ||\mathcal{E}_L||$ if and only if $||\mathcal{D}||$ and $||\mathcal{D}_2||$ agree on L which is undecidable. □

4.2. Isomorphism

For a $\{0,1\}$-weighted automaton $\mathcal{A}$ over an ordered alphabet $(\Gamma;\leq)$, we define a linear order $\mathcal{L}_\mathcal{A}$ setting

$$\mathcal{L}_\mathcal{A} = \sum_{w\in(\Gamma^+;\leq_{\text{llex}})} (\omega^{||\mathcal{A}||(w)+1} + \delta)$$

(if $||\mathcal{A}||(w) = -\infty$, then we define $\omega^{||\mathcal{A}||(w)+1}$ as the empty set $\underline{0}$). Since $(\Gamma^+;\leq_{\text{llex}}) \cong \omega$, this linear order is an ω-sequence of ordinals ω^n with $n \geq 1$ and $\underline{0}$, separated by our delimiter δ. Hence it is scattered. Furthermore, we obtain

Lemma 4.9. *Let $\mathcal{A}$ and $\mathcal{B}$ be $\{0,1\}$-weighted automata. Then $\mathcal{L}_\mathcal{A} \cong \mathcal{L}_\mathcal{B}$ if and only if $||\mathcal{A}|| = ||\mathcal{B}||$.*

PROOF. The implication "$\Leftarrow$" is trivial by the very definition of $\mathcal{L}_\mathcal{A}$. So let f be an isomorphism from $\mathcal{L}_\mathcal{A}$ onto $\mathcal{L}_\mathcal{B}$. Note that the intervals of type $\delta \cong \omega\cdot\omega^*$ in $\mathcal{L}_\mathcal{A}$ form an ω-chain. Hence, the isomorphism f has to map the n^{th} such interval in $\mathcal{L}_\mathcal{A}$ onto the n^{th} such interval in $\mathcal{L}_\mathcal{B}$. Consequently, $\omega^{||\mathcal{A}||(w)+1} \cong \omega^{||\mathcal{B}||(w)+1}$ implying $||\mathcal{A}||(w) = ||\mathcal{B}||(w)$ for all $w \in \Gamma^+$. □

Lemma 4.10. *From a $\{0,1\}$-weighted automaton $\mathcal{A}$, one can compute a regular tree-language $L_\mathcal{A}$ such that $(L_\mathcal{A};\leq_{\text{trees}}) \cong \mathcal{L}_\mathcal{A}$.*

Before we prove this lemma, we show how we can use it to prove that the isomorphism problem of scattered tree-automatic linear orders is undecidable (the proof of Lemma 4.10 can be found following Corollary 4.13).

Theorem 4.11. *There is a scattered linear order $\mathcal{L}$ such that the set of regular tree-languages L with $(L; \leq_{\text{trees}}) \cong \mathcal{L}$ is Π^0_1-hard.*

Proof. Let $\mathcal{C}$ be the $\{0,1\}$-weighted automaton from Theorem 4.8 and set $\mathcal{L} = \mathcal{L}_{\mathcal{C}}$. Furthermore, let f denote the computable function that maps a $\{0,1\}$-weighted automaton $\mathcal{E}$ to the regular tree-language $L_{\mathcal{E}}$ (cf. Lemma 4.10). Then, by Lemmas 4.9 and 4.10, f is a reduction from the Π^0_1-complete set of $\{0,1\}$-weighted automata $\mathcal{E}$ with $||\mathcal{E}|| = ||\mathcal{C}||$ to the set of regular tree-languages L with $\mathcal{L} \cong (L; \leq_{\text{trees}})$. □

Since the linear order $\leq_{\text{trees}}$ is tree-automatic, we immediately obtain

Corollary 4.12. *There is a scattered linear order $\mathcal{L}$ whose set of tree-automatic presentations is Π^0_1-hard.*

From this, we can infer that the isomorphism problem for tree-automatic scattered linear orders is Π^0_1-hard. We do not know whether the set of tree-automatic presentations of *scattered* linear orders is decidable. Therefore, the formulation of the following immediate consequence of Corollary 4.12 is a bit involved:

Corollary 4.13. *Let X be a set of pairs of tree-automatic presentations such that, for all tree-automatic presentations P_1 and P_2 of scattered linear orders $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively, one has*

$$(P_1, P_2) \in X \iff \mathcal{L}_1 \cong \mathcal{L}_2 \,.$$

Then X is Π^0_1-hard.

The rest of this section is devoted to the proof of Lemma 4.10.

Proof of Lemma 4.10. Let $\mathcal{A} = (Q, \Gamma, \iota, \mu, F)$ be a $\{0,1\}$-weighted automaton. We will construct a tree-automatic presentation of the linear order $\mathcal{L}_{\mathcal{A}}$.

A *run tree of* $\mathcal{A}$ (cf. Fig. 4 for an example where we omitted the label \$) is a tree t over the alphabet $\Gamma \uplus \{\$\}$ such that there exists a sequence of states $\iota = q_0, q_1, \ldots, q_{k-1} \in Q$ and $q_k \in F$ (with $k = \max\{i \mid 0^{i+1} \in \text{dom}(t)\}$) with the following properties:

(T1) $11 \in \text{dom}(t) \subseteq 0^* \cup 0^*10^* \cup 110^*$ and $100 \notin \text{dom}(t)$

(T2) $t(0^i) \in \Gamma$ and $\mu(q_{i-1}, t(0^i), q_i) \neq -\infty$ for all $1 \leq i \leq k$

(T3) $0^i1 \in \text{dom}(t)$ implies $i = 0$ or $1 \leq i \leq k$ and $\mu(q_{i-1}, a_i, q_i) = 1$

(T4) $t^{-1}(\$) = \text{dom}(t) \setminus \{0^i \mid 1 \leq i \leq k\}$

Note that every run tree t defines a word over Γ, namely

$$\text{word}(t) = t(0)\, t(00)\, \ldots\, t(0^k)\,.$$

Since $11 \in \text{dom}(t)$, also 1 and therefore 0 belong to $\text{dom}(t)$ and therefore $\text{word}(t) \neq \varepsilon$ (the run tree t from Fig. 4 satisfies $\text{word}(t) = abaab$). The idea is that the "main branch" $\{0, 00, \ldots, 0^k\}$ carries a run ρ of the weighted automaton $\mathcal{A}$. The number of "side branches" starting in some node 0^i1 with $i > 0$ is at most the weight $\text{wt}(\rho)$ of the encoded run. Since these side branches have arbitrary length, the whole run tree stands

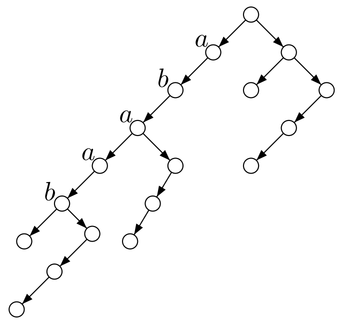

Figure 4: A run tree

for an element of $\omega^{\mathrm{wt}(\rho)}$. The "side branch" starting in 11 plays a special role, its length $|\mathrm{dom}(t) \cap 110^+|$ is denoted $n(t)$ (the run tree t in Fig. 4 satisfies $n(t) = 2$).

We next define, for two trees s and t, the tree $s + t$ by adding a new \$-labeled root and considering s as left subtree of $s + t$ and t as right subtree. More formally, $\mathrm{dom}(s+t) = \{\varepsilon\} \cup 0\,\mathrm{dom}(s) \cup 1\,\mathrm{dom}(t)$, $(s+t)(\varepsilon) = \$$, $(s+t)(0\,u) = s(u)$ for $u \in \mathrm{dom}(s)$, and $(s+t)(1\,v) = t(v)$ for $v \in \mathrm{dom}(t)$. Since we consider words as special trees, we will meet trees of the form $w + t$. These trees carry the sequences $\$w$ on $\mathrm{dom}(w+t) \cap 0^*$ and satisfy $(w + t)|_1 \cong t$.

We now define the language $L_{\mathcal{A}}$ by

$$L_{\mathcal{A}} = \{t \mid t \text{ is a run tree}\} \cup \{w\$ + t \mid w \in \Gamma^+, t \in D\}$$

where D is the set of trees from page 4 that satisfies $(D; \leq_{\mathrm{trees}}) \cong \delta$. This language is clearly regular.

Note that trees from $L_{\mathcal{A}}$ use, besides letters from Γ, the letter \$; we order $\Gamma \cup \{\$\}$ in such a way that the order on Γ is preserved. We will now prove

$$(L_{\mathcal{A}}; \leq_{\mathrm{trees}}) \cong \mathcal{L}_{\mathcal{A}}\,.$$

First let $\rho = (q_0, a_1, q_1)(q_1, a_2, q_2) \ldots (q_{k-1}, a_k, q_k) \in \mathrm{Run}(\mathcal{A}, w)$ be a run of the weighted automaton $\mathcal{A}$ on the word $w = a_1 \ldots a_k$. For $n \in \mathbb{N}$, let $\mathcal{I}^0_{\rho,n}$ denote the restriction of $(L_{\mathcal{A}}; \leq_{\mathrm{trees}})$ to all run trees t with $\mathrm{word}(t) = w$, $n(t) = n$, and such that (T2) and (T3) hold with the sequence of states $q_0, q_1, \ldots, q_k$. For any tuple $(m_1, \ldots, m_k) \in \mathbb{N}^k$ such that

$$m_i > 0 \implies \mu(q_{i-1}, a_i, q_i) = 1\,,$$

there exists a unique run tree $t \in \mathcal{I}^0_{\rho,n}$ satisfying $|\mathrm{dom}(t) \cap 0^i 10^*| = m_i$ for all $1 \leq i \leq k$. Conversely, by (T3), any run tree from $\mathcal{I}^0_{\rho,n}$ arises this way. Hence we get

$$\mathcal{I}^0_{\rho,n} \cong \omega^{\mathrm{wt}(\rho)}\,.$$

Next let $w \in \Gamma^+$ and $n \in \mathbb{N}$. Then $\mathcal{I}^1_{w,n}$ denotes the restriction of $(L_{\mathcal{A}}; \leq_{\mathrm{trees}})$ to all run trees t with

$$\mathrm{word}(t) = w \text{ and } n(t) = n\,. \tag{2}$$

In other words, $\mathcal{I}^1_{w,n}$ is the union of the linear orders $\mathcal{I}^0_{\rho,n}$ over all runs ρ on the word w (note that this union is not necessarily disjoint). Let ρ be the run on w of maximal weight. Then

$$\omega^{||\mathcal{A}||(w)} = \omega^{\mathrm{wt}(\rho)} \leq \mathcal{I}^1_{w,n} \,.$$

Let H be the disjoint union of the sets $\mathcal{I}^0_{\rho,n}$, i.e., the set of pairs (t,ρ) where ρ is a run of the weighted automaton $\mathcal{A}$ on w and $t \in \mathcal{I}^0_{\rho,n} \subseteq \mathcal{I}^1_{w,n}$. We order this set setting $(t_1,\rho_1) \leq (t_2,\rho_2)$ if and only if

$$t_1 <_{\mathrm{trees}} t_2 \text{ or } t_1 = t_2 \text{ and } \rho_1 \leq_{\mathrm{lex}} \rho_2 \,.$$

Then we have

$$\begin{aligned}
\mathcal{I}^1_{w,n} &= \Big(\bigcup_{\rho\in\mathrm{Run}(\mathcal{A},w)} \mathcal{I}^0_{\rho,n}; \leq_{\mathrm{trees}}\Big) \\
&\leq (H;\leq) \\
&\leq \bigoplus_{\rho\in\mathrm{Run}(\mathcal{A},w)} (\mathcal{I}^0_{\rho,n}; \leq_{\mathrm{trees}}) \\
&= \bigoplus_{\rho\in\mathrm{Run}(\mathcal{A},w)} \omega^{\mathrm{wt}(\rho)} \\
&\leq \omega^{||\mathcal{A}||(w)} \cdot \underline{|\mathrm{Run}(\mathcal{A},w)|}
\end{aligned}$$

where $\oplus$ denotes the natural sum of ordinals.

In summary, we have

$$\begin{aligned}
\omega^{||\mathcal{A}||(w)+1} \leq \mathcal{I}^1_{w,n} \cdot \omega &\leq \Big(\omega^{||\mathcal{A}||(w)} \cdot \underline{|\mathrm{Run}(\mathcal{A},w)|}\Big) \cdot \omega \\
&= \omega^{||\mathcal{A}||(w)+1}
\end{aligned}$$

and therefore

$$\mathcal{I}^1_{w,n} \cdot \omega = \omega^{||\mathcal{A}||(w)+1} \,.$$

Next consider the restriction $\mathcal{I}^1_w$ of $(L_{\mathcal{A}}; \leq_{\mathrm{trees}})$ to the set of run trees t with $\mathrm{word}(t) = w$. Then $n(s) < n(t)$ implies $s <_{\mathrm{trees}} t$. Furthermore, the restriction of $\mathcal{I}^1_w$ to the set of run trees t with $n(t) = n$ equals $\mathcal{I}^1_{w,n}$. Note also that $\mathcal{I}^1_{w,0} \cong \mathcal{I}^1_{w,n}$ for all $n \geq 0$. Hence

$$\mathcal{I}^1_w = \sum_{n\in(\mathbb{N};\leq)} \mathcal{I}^1_{w,n} = \mathcal{I}^1_{w,0} \cdot \omega = \omega^{||\mathcal{A}||(w)+1} \,.$$

Next consider the restriction $\mathcal{I}^2_w$ of $(L_{\mathcal{A}}; \leq_{\mathrm{trees}})$ to the set of trees $w\$ + D$. Then $\mathcal{I}^2_w \cong \delta$ by what we saw on page 4. Let s be a run tree with $\mathrm{word}(s) = w$ and let $t \in w\$ + D$. Then s and t coincide on 0^* (where they both carry the sequence $\$w\$$). Consider $s{\restriction}_{10^*}$ and $t{\restriction}_{10^*}$. Since s is a run tree, we have $\mathrm{dom}(s) \cap 10^* = \{1, 10\}$ while $t|_1 \in D$ implies $\mathrm{dom}(t) \cap 10^* = \{1, 10, 100\}$. Hence $s|_1 <_{\mathrm{trees}} t|_1$ and therefore $s <_{\mathrm{trees}} t$. Hence, the restriction $\mathcal{I}_w$ of $(L_{\mathcal{A}}; \leq_{\mathrm{trees}})$ to the set of run trees t with $\mathrm{word}(t) = w$ and the set of trees $w\$ + D$ satisfies

$$\mathcal{I}_w = \mathcal{I}^1_w + \mathcal{I}^2_w \cong \omega^{||\mathcal{A}||(w)+1} + \delta \,.$$

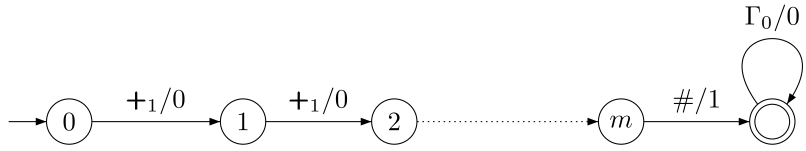

Figure 5: The automaton $\mathcal{C}_m$ from the proof of Lemma 4.14

Finally, let $u, v \in \Gamma^+$, $s \in \mathcal{I}_u$ and $t \in \mathcal{I}_v$. Then $u <_{\text{llex}} v$ if and only if $s <_{\text{trees}} t$. This implies

$$(L_{\mathcal{A}}; \leq_{\text{trees}}) = \sum_{w \in (\Gamma^+; \leq_{\text{llex}})} \mathcal{I}_w \cong \sum_{w \in (\Gamma^+; \leq_{\text{llex}})} \omega^{||\mathcal{A}||(w)+1} + \delta = \mathcal{L}_{\mathcal{A}} \,.$$

□

4.3. Automorphisms

From Theorem 3.4, we already know that the existence of a non-trivial automorphism of a word-automatic and scattered linear order is Σ^0_1-hard. Here, we push this lower bound one level higher for tree-automatic scattered linear orders. The order theoretic construction resembles that from Section 3.1, but also uses ideas from the previous section.

The general strategy of proof is to construct, from a two-counter machine M, a tree-automatic linear order $\mathcal{L}_M$ such that $\mathcal{L}_M$ is rigid if and only if the halting set of M equals $\mathbb{N}$. Since this problem is Π^0_2-complete, we obtain that the existence of a nontrivial automorphism is Σ^0_2-hard.

But first, we need another lemma about weighted automata:

Lemma 4.14. *From a two-counter machine M and $m \in \mathbb{N}$, one can construct a $\{-1, 0, 1\}$-weighted automaton $\mathcal{C}_{M,m}$ over Γ such that the following are equivalent:*

- *m is not accepted by M*
- *$||\mathcal{C}_{M,m}|| = ||\mathcal{C}_M||$ where $\mathcal{C}_M$ is the weighted automaton from Lemma 4.4.*

Proof. Let $m \in \mathbb{N}$ and consider the $\{-1, 0, 1\}$-weighted automaton $\mathcal{C}_m$ in Fig. 5. Then

$$||\mathcal{C}_m||(w) = \begin{cases} 1 & \text{if } w \in {+_1}^m \# \Gamma_0^* \\ -\infty & \text{otherwise.} \end{cases}$$

Note that the $\{-1, 0, 1\}$-weighted automaton $\mathcal{C}_m$ can be constructed from m.

Next let $\mathcal{C}_{M,m}$ be the disjoint union of $\mathcal{C}_M$ and $\mathcal{C}_m$ such that

$$||\mathcal{C}_{M,m}||(w) = \max(||\mathcal{C}_M||(w), ||\mathcal{C}_m||(w)) \,.$$

Then we have $||\mathcal{C}_{M,m}||(w) = ||\mathcal{C}_M||(w)$ if and only if $||\mathcal{C}_m||(w) \leq ||\mathcal{C}_M||(w)$. This is the case if and only if $||\mathcal{C}_m||(w) = -\infty$ or $||\mathcal{C}_M||(w) > 0$. But this is equivalent to saying

$$\text{if } w = {+_1}^m \# u \text{ with } m \in \mathbb{N} \text{ and } u \in \Gamma_0^*, \quad \text{then } u \text{ is not a halting computation of } M \text{ from } m.$$

Hence $||\mathcal{C}_{M,m}|| = ||\mathcal{C}_M||$ if and only if there is no halting computation of M from m. □

So let M be a two-counter machine. Then, let $\mathcal{C}_M$ be the $\{-1,0,1\}$-weighted automaton from Lemma 4.4 and, for $m \in \mathbb{N}$, let $\mathcal{C}_{M,m}$ be the $\{-1,0,1\}$-weighted automaton from Lemma 4.14. Furthermore, let $\mathcal{D}_M$ and $\mathcal{D}_{M,m}$ be the $\{0,1\}$-weighted automata constructed using Lemma 4.6 from $\mathcal{C}_M$ and $\mathcal{C}_{M,m}$, respectively.

Then we define the linear order

$$\mathcal{L}_M = \sum_{m \in (\mathbb{N};\leq)} \left(\mathcal{L}_{\mathcal{D}_M} \cdot \omega^* + \mathcal{L}_{\mathcal{D}_{M,m}} \cdot \omega\right) .$$

Lemma 4.15. *Let M be a two-counter machine. Then $\mathcal{L}_M$ is rigid if and only if the halting set of M equals $\mathbb{N}$.*

PROOF. Let $\mathcal{D}_M$ and $\mathcal{D}_{M,m}$ be the $\{0,1\}$-weighted automata from above.

First suppose there is some $m \in \mathbb{N}$ such that the two-counter machine M does not accept m. Then $||\mathcal{C}_{M,m}|| = ||\mathcal{C}_M||$ by Lemma 4.14 and therefore $||\mathcal{D}_{M,m}|| = ||\mathcal{D}_M||$ by Lemma 4.6. Hence $\mathcal{L}_M$ contains an interval of the form

$$\mathcal{L}_{\mathcal{D}_{M,m}} \cdot \omega^* + \mathcal{L}_{\mathcal{D}_{M,m}} \cdot \omega = \mathcal{L}_{\mathcal{D}_{M,m}} \cdot \zeta \,.$$

Since ζ is not rigid, this interval and therefore $\mathcal{L}_M$ has a nontrivial automorphism. This proves the implication "$\Rightarrow$".

For the other implication let f be a nontrivial automorphism of $\mathcal{L}_M$. Note that

$$\mathcal{L}_M = \sum_{m \in (\mathbb{N};\leq)} \left(\begin{array}{c} \left(\sum_{w \in (\Gamma^+,\leq_{\text{llex}})} \left(\omega^{||\mathcal{D}_M||(w)+1} + \omega^* \cdot \omega\right)\right) \cdot \omega^* \\ + \left(\sum_{w \in (\Gamma^+,\leq_{\text{llex}})} \left(\omega^{||\mathcal{D}_{M,m}||(w)+1} + \omega^* \cdot \omega\right)\right) \cdot \omega \end{array} \right) .$$

Let $\mathcal{L}'$ be the set of intervals of type $\omega^* \cdot \omega$ with the order inherited from $\mathcal{L}_M$. Then f induces an automorphism f' of

$$\mathcal{L}' \cong \sum_{m \in (\mathbb{N};\leq)} (\omega \cdot \omega^* + \omega \cdot \omega) \cong \sum_{m \in (\mathbb{N};\leq)} \omega \cdot \zeta \,.$$

Note that every maximal interval of $\mathcal{L}_M$ not intersecting any copy of $\omega^* \cdot \omega$ is an ordinal and therefore rigid. Hence f' is nontrivial.

Next let $\sim$ be the equivalence relation on $\mathcal{L}'$ with $x \sim y$ if there are only finitely many elements in between x and y. Then every $\sim$-equivalence classes in $\mathcal{L}'$ is isomorphic to ω and

$$\mathcal{L}'' = \mathcal{L}'/{\sim} \cong \sum_{m \in (\mathbb{N};\leq)} \zeta \,.$$

Furthermore, f' induces an automorphism f'' of $\mathcal{L}''$. Since all $\sim$-equivalence classes in $\mathcal{L}'$ are rigid, the automorphism f'' is nontrivial. Note that f'' maps every interval of type ζ onto itself. Hence there is $m \in \mathbb{N}$ such that f'' acts nontrivially on the m^{th} copy of ζ. Consequently, f' moves some interval of type ω in the m^{th} copy of $\omega \cdot \zeta = \omega \cdot \omega^* + \omega \cdot \omega$ to some other interval of type ω in this copy. We can assume that it maps the last interval of type ω in $\omega \cdot \omega^*$ to some copy of ω in $\omega \cdot \omega$. Consequently f maps the last copy of $\mathcal{L}_{\mathcal{D}_M}$

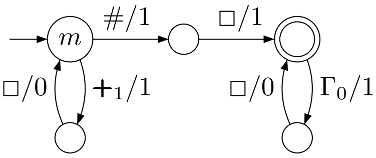

Figure 6: The automaton $\mathcal{D}$ from the proof of Lemma 4.16

to some copy of $\mathcal{L}_{\mathcal{D}_{M,m}}$ in

$$\begin{aligned}\mathcal{L}_m = & \left(\sum_{w\in(\Gamma^+,\leq_{\text{llex}})}(\omega^{||\mathcal{D}_M||(w)+1} + \omega^* \cdot \omega)\right) \cdot \omega^* \\ & + \left(\sum_{w\in(\Gamma^+,\leq_{\text{llex}})}(\omega^{||\mathcal{D}_{M,m}||(w)+1} + \omega^* \cdot \omega)\right) \cdot \omega \\ & = \mathcal{L}_{\mathcal{D}_M} \cdot \omega^* + \mathcal{L}_{\mathcal{D}_{M,m}} \cdot \omega\,.\end{aligned}$$

This implies $\mathcal{L}_{\mathcal{D}_M} \cong \mathcal{L}_{\mathcal{D}_{M,m}}$. From Lemma 4.14, we obtain that m is not accepted by M. Hence the halting set of M is not $\mathbb{N}$. This finishes the proof of the implication "$\Leftarrow$". □

Lemma 4.16. *From a two-counter machine M, one can construct a tree-automatic presentation of the linear order $\mathcal{L}_M$.*

Proof. So let M be a two-counter machine. Let $\mathcal{C}_M$ and $\mathcal{C}_{M,m}$ be the $\{-1,0,1\}$-weighted automata from Lemmas 4.4 and 4.14 and let $\mathcal{D}_M$ and $\mathcal{D}_{M,m}$ be the results of applying Lemma 4.6 to these weighted automata. We will also need the $\{-1,0,1\}$-weighted automaton $\mathcal{C}_m$ from the proof of Lemma 4.14 and denote $\mathcal{D}_m$ the result of applying Lemma 4.6 to this weighted automaton.

Next, we define the tree-language L_M that will serve as universe of the tree-automatic copy of $\mathcal{L}_M$:

$$\begin{aligned}L_M = & \ L_{\mathcal{D}_M} \otimes \$^* \otimes €(+_1\Box)^* \\ & \cup L_{\mathcal{D}_M} \otimes \$^* \otimes \$(+_1\Box)^* \\ & \cup \{t \otimes \$^k \otimes \$(+_1\Box)^m \mid k,m \geq 0, t \text{ is a run tree of } \mathcal{D}_m\}\end{aligned}$$

Since the set $L_{\mathcal{D}_M}$ is regular, so are the two first subsets of L_M.

To prove the regularity of the third tree-language, we make use of the weighted automaton $\mathcal{D}$ from Fig. 6. Let $m \in \mathbb{N}$ and let t be a tree. Then t is a run tree of $\mathcal{D}_m$ if and only if t is a run tree of $\mathcal{D}$ with $\text{word}(t) \in \$(+_1\Box)^m\#\Box(\Gamma_0\Box)^*$. In other words, the third tree-language equals the set of trees $t \otimes \$^k \otimes \$(+_1\Box)^m$ with $k,m \geq 0$ such that t is a run tree of $\mathcal{D}$ with $\text{word}(t) \in \$(+_1\Box)^m\#\Box(\Gamma_0\Box)^*$. Since the set of run trees of $\mathcal{D}$ is regular and since a tree automaton running on the convolution of a tree and two words can compare the main branch of the tree and the third word, also the third subset of L_M is regular. Hence, indeed, L_M is even effectively regular.

Next we define a linear order $\preceq$ on L_M. We set $(s\otimes\$^k\otimes c(+_1\Box)^m) \preceq (t\otimes\$^\ell\otimes d(+_1\Box)^n)$ (with $k,\ell,m,n \in \mathbb{N}$ and $c,d \in \{€,\$\}$) if and only if we have

(O1) $m < n$, or

(O2) $m = n$, $c = €$, and $d = \$$, or

(O3) $m = n$, $c = d = €$, and $k > \ell$, or

(O4) $m = n$, $c = d = €$, $k = \ell$, and $s \leq_{\text{trees}} t$, or

(O5) $m = n$, $c = d = \$$, and $k < \ell$, or

(O6) $m = n$, $c = d = \$$, $k = \ell$, and $s \leq_{\text{trees}} t$.

It is clear that this relation is automatic and it remains to be shown that $(L_M; \preceq) \cong \mathcal{L}_M$.

For $k, m \geq 0$ let $\mathcal{I}^1_{k,m}$ denote the restriction of $(L_M; \preceq)$ to the set $L_{\mathcal{D}_M} \otimes \$^k \otimes €(a\Box)^m$. By (O4) and Lemma 4.10, we get

$$\mathcal{I}^1_{k,m} \cong \mathcal{L}_{\mathcal{D}_M} \, . \tag{3}$$

Next let $\mathcal{I}^1_m$ denote the restriction of $(L_M; \preceq)$ to the set $L_{\mathcal{D}_M} \otimes \$^* \otimes €(a\Box)^m$. Then, (O3) and (3) imply

$$\mathcal{I}^1_m \cong \mathcal{L}_{\mathcal{D}_M} \cdot \omega^* \, . \tag{4}$$

On the other hand, for $k, m \geq 0$, let $\mathcal{I}^2_{k,m}$ denote the restricion of $(L_M; \preceq)$ to the set

$$L_{\mathcal{D}_M} \otimes \$^k \otimes \$(+_1\Box)^m \cup \{t \otimes \$^k \otimes \$(+_1\Box)^m \mid t \text{ is a run tree of } \mathcal{D}_m\} \, .$$

By the definition of $L_{\mathcal{D}_M}$, this set equals

$$\begin{aligned} &\{(w\$ + t) \otimes \$^k \otimes \$(+_1\Box)^m \mid w \in \Gamma^+ \text{ and } t \in D\} \\ \cup &\{t \otimes \$^k \otimes \$(+_1\Box)^m \mid t \text{ is a run tree of } \mathcal{D}_m \text{ or } \mathcal{D}_M\} \, . \end{aligned}$$

Recall from the proof of Lemma 4.14 that $\mathcal{C}_{M,m}$ is the disjoint union of the $\{-1, 0, 1\}$-weighted automata $\mathcal{C}_m$ and $\mathcal{C}_M$. Consequently, $\mathcal{D}_{M,m}$ is the disjoint union of the $\{0, 1\}$-weighted automata $\mathcal{D}_m$ and $\mathcal{D}_M$. Hence the above set equals $L_{\mathcal{D}_{M,m}}$. Now, from (O6) and Lemma 4.14, we obtain

$$\mathcal{I}^2_{k,m} \cong \mathcal{L}_{\mathcal{D}_{M,m}} \, . \tag{5}$$

Next let $\mathcal{I}^2_m$ denote the restriction of $(L_M; \preceq)$ to the set

$$L_{\mathcal{D}_M} \otimes \$^* \otimes \$(+_1\Box)^m \cup \{t \otimes \$^k \otimes \$(+_1\Box)^m \mid k \geq 0, t \text{ is a run tree of } \mathcal{D}_m\} \, .$$

Then, (O5) and (5) imply

$$\mathcal{I}^1_m \cong \mathcal{L}_{\mathcal{D}_M} \cdot \omega \, . \tag{6}$$

Now, from (O2), (4) and (6), we obtain that the restriction of $(L_M; \preceq)$ to the set of trees that define $\mathcal{I}^1_m$ and $\mathcal{I}^2_m$ is isomorphic to

$$\mathcal{L}_{\mathcal{D}_M} \cdot \omega^* + \mathcal{L}_{\mathcal{D}_{M,m}} \cdot \omega \, .$$

Finally, (O1) implies

$$\begin{aligned} (L_M; \preceq) &\cong \sum_{m \in (\mathbb{N}; \leq)} \mathcal{L}_{\mathcal{D}_M} \cdot \omega^* + \mathcal{L}_{\mathcal{D}_{M,m}} \cdot \omega \\ &= \mathcal{L}_M \, . \end{aligned}$$

$\Box$

Theorem 4.17. *(i) The set of tree-automatic presentations of rigid (rigid and scattered, resp.) linear orders is Π^0_2-hard.*

(ii) The set of tree-automatic presentations of automatically rigid linear orders is Π^0_1-complete.

Proof. (i) Let F be the set of two-counter machines whose halting set is $\mathbb{N}$. Then, by Theorem 4.3, F is Π^0_2-complete. Lemmas 4.15 and 4.16 reduce F to the set of rigid (and scattered) tree-automatic linear orders.

(ii) Hardness follows from Corollary 3.5(ii), containment in Π^0_1 can be shown as in the proof of Corollary 3.5(ii). □

It follows in particular that there exists a tree-automatic scattered linear order that has non-trivial automorphisms, but no tree-automatic non-trivial automorphisms.

5. Open questions

The isomorphism and rigidity problems for word-automatic scattered linear orders both belong to Δ^0_ω (cf. [25]), our lower bound Π^0_1 for the rigidity problem leaves quite some room for improvements. Since the rank of a tree-automatic linear order is properly below ω^ω [15], the proof of [25, Theorem 5.19] can be adapted to show that the isomorphism and the rigidity problems for tree-automatic scattered linear orders both belong to $\Delta^0_{\omega^\omega}$. But we only have the lower bounds Π^0_1 and Π^0_2, resp. Finally, the rigidity problem for arbitrary word- or tree-automatic linear orders is in Π^1_1, but also here, we only have the arithmetic lower bound Π^0_1 and Π^0_2, resp.

But the most pressing open question is the isomorphism problem of scattered and word-automatic linear orders.

References

[1] S. Almagor, U. Boker, and O. Kupferman. What's decidable about weighted automata? In *ATVA'11*, Lecture Notes in Comp. Science vol. 6996, pages 482–491. Springer, 2011.

[2] V. Bárány, E. Grädel, and S. Rubin. Automata-based presentations of infinite structures. In *Finite and Algorithmic Model Theory*, pages 1–76. Cambridge University Press, 2011.

[3] A. Blumensath and E. Grädel. Automatic Structures. In *LICS'00*, pages 51–62. IEEE Computer Society Press, 2000.

[4] A. Blumensath and E. Grädel. Finite presentations of infinite structures: Automata and interpretations. *Theory of Computing Systems*, 37(6):641–674, 2004.

[5] G. Braun and L. Strüngmann. Breaking up finite automata presentable torsion-free abelian groups. *International Journal of Algebra and Computation*, 21(8):1463–1472, 2011.

[6] A. Carayol and Z. Ésik. A context-free linear ordering with an undecidable first-order theory. In *IFIP-TCS'12*, Lecture Notes in Comp. Science vol. 7604, pages 104–118. Springer, 2012.

[7] A. Carayol and Z. Ésik. The FC-rank of a context-free language. *Inf. Process. Lett.*, 8:285–287, 2013.

[8] Ch. Delhommé. Automaticité des ordinaux et des graphes homogènes. *C. R. Acad. Sci. Paris, Ser. I*, 339:5–10, 2004.

[9] M. Droste and D. Kuske. Weighted Automata. To appear in Handbook AutoMathA, 2013.

[10] A. Durand-Gasselin and P. Habermehl. Ehrenfeucht-Fraïssé goes elementarily automatic for structures of bounded degree. In *STACS'12*, pages 242–253. Dagstuhl Publishing, 2012.

[11] C.C. Elgot. Decision problems of finite automata design and related arithmetics. *Trans. Am. Math. Soc.*, 98:21–51, 1961.

[12] D.B.A. Epstein, J.W. Cannon, D.F. Holt, S.V.F. Levy, M.S. Paterson, and W.P. Thurston. *Word Processing In Groups.* Jones and Bartlett Publishers, Boston, 1992.

[13] Z. Ésik. An undecidable property of context-free linear orders. *Inform. Processing Letters*, 111(3):107–109, 2011.

[14] B.R. Hodgson. On direct products of automaton decidable theories. *Theoretical Computer Science*, 19:331–335, 1982.

[15] M. Huschenbett. The rank of tree-automatic linear orderings. In *STACS '13*, Leibniz International Proceedings in Informatics (LIPIcs) vol. 20, pages 586–597. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2013.

[16] S. Jain, B. Khoussainov, Ph. Schlicht, and F. Stephan. Tree-automatic scattered linear orders. accepted for Proceedings of the Institute of Mathematical Sciences, National University of Singapore, 2013.

[17] A. Kach and M. Minnes. Automatic linear orders. Submitted, 2013.

[18] O. Kharlampovich, B. Khoussainov, and A. Miasnikov. From automatic structures to automatic groups. arXiv:0810.0935v2, 2011.

[19] B. Khoussainov and A. Nerode. Automatic presentations of structures. In *Logic and Computational Complexity*, Lecture Notes in Comp. Science vol. 960, pages 367–392. Springer, 1995.

[20] B. Khoussainov and A. Nerode. Open questions in the theory of automatic structures. *Bulletin of the EATCS*, 94:181–204, 2008.

[21] B. Khoussainov, A. Nies, S. Rubin, and F. Stephan. Automatic structures: richness and limitations. *Log. Methods in Comput. Sci.*, 3(2), 2007.

[22] B. Khoussainov, S. Rubin, and F. Stephan. Automatic linear orders and trees. *ACM Transactions on Computational Logic*, 6(4):675–700, 2005.

[23] D. Krob. The equality problem for rational series with multiplicities in the tropical semiring is undecidable. *International Journal of Algebra and Computation*, 4(3):405–425, 1994.

[24] D. Kuske. Logical aspects of the lexicographic order on 1-counter languages. In *MFCS'13*, Lecture Notes in Comp. Science vol. 3114, pages 619–630. Springer, 2013.

[25] D. Kuske, J. Liu, and M. Lohrey. The isomorphism problem on classes of automatic structures with transitive relations. *Transactions of the AMS*, 365:5103–5151, 2013.

[26] J. Liu and M. Minnes. Deciding the isomorphism problem in classes of unary automatic structures. *Theoretical Computer Science*, 412(18):1705–1717, 2011.

[27] M. Lohrey and Ch. Mathissen. Isomorphism of regular trees and words. *Information and Computation*, 224:71–105, 2013.

[28] Y. Matijasevich. *Hilbert's Tenth Problem.* Foundations of Computing Series. MIT Press, 1993.

[29] A. Miasnikov and Zoran Šunić. Cayley graph automatic groups are not necessarily Cayley graph biautomatic. In *LATA'12*, Lecture Notes in Comp. Science vol. 7183, pages 401–407. Springer, 2012.

[30] M. Minsky. Recursive unsolvability of Post's problem of 'tag' and other topics in theory of Turing machines. *Annals of Mathematics*, 74(3):437–455, 1961.

[31] A. Nies. Describing groups. *Bulletin of Symbolic Logic*, 13(3):305–339, 2007.

[32] S. Rubin. *Automatic Structures.* PhD thesis, University of Auckland, 2004.

[33] S. Rubin. Automata presenting structures: A survey of the finite string case. *Bulletin of Symbolic Logic*, 14:169–209, 2008.

[34] J. Sakarovitch. Rational and recognizable series. In M. Droste, W. Kuich, and H. Vogler, editors, *Handbook of Weighted Automata*, pages 405–453. Springer, 2009.