Tree-Automatic Well-Founded Trees

Alexander Kartzow[1], Jiamou Liu[2], and M. Lohrey[1, ⋆]

[1] Universität Leipzig, Germany
[2] Auckland University of Technology, New Zealand
kartzow@informatik.uni-leipzig.de, jiamou.liu@aut.ac.nz,
lohrey@informatik.uni-leipzig.de

Abstract. We investigate tree-automatic well-founded trees. For this, we introduce a new ordinal measure for well-founded trees, called ∞-rank. The ∞-rank of a well-founded tree is always bounded from above by the ordinary (ordinal) rank of a tree. We also show that the ordinal rank of a well-founded tree of ∞-rank α is smaller than $\omega \cdot (\alpha + 1)$. For string-automatic well-founded trees, it follows from [16] that the ∞-rank is always finite. Here, using Delhommé's decomposition technique for tree-automatic structures, we show that the ∞-rank of a tree-automatic well-founded tree is strictly below ω^ω. As a corollary, we obtain that the ordinal rank of a string-automatic (resp., tree-automatic) well-founded tree is strictly below ω^2 (resp., ω^ω). The result for the string-automatic case nicely contrasts a result of Delhommé, saying that the ranks of string-automatic well-founded partial orders reach all ordinals below ω^ω. As a second application of the ∞-rank we show that the isomorphism problem for tree-automatic well-founded trees is complete for level $\Delta^0_{\omega^\omega}$ of the hyperarithmetical hierarchy (under Turing-reductions). Full proofs can be found in the arXiv-version [11] of this paper.

1 Introduction

Various classes of infinite but finitely presented structures received a lot of attention in algorithmic model theory [2]. Among the most important such classes of structures is the class of *string-automatic structures* [13]. A (relational) structure is string-automatic if its universe is a regular set of words and all relations can be recognized by synchronous multi-tape automata. During the past 15 years a theory of string-automatic structures has emerged. This theory was developed along two interrelated branches. The first is a structural branch, which leads to (partial) characterizations of particular classes of string-automatic structures [6,12,14,15,18]. The second is an algorithmic branch, which leads to numerous decidability and undecidability, as well as complexity results for important algorithmic problems for string-automatic structures [4,14,17]. One of the most fundamental results for string-automatic structures states that their first-order theories are uniformly decidable [13].

By replacing strings and string automata by trees and tree automata, Blumensath [3] generalized string-automatic structures to *tree-automatic structures* and proved that their first-order theories are still uniformly decidable. However compared to string-automatic structures, the theory of tree-automatic structures is less developed. The only

⋆ The first and third author are supported by the DFG research project GELO.

non-trivial characterization of a class of tree-automatic structures we are aware of concerns ordinals. Delhommé proved in [6] that an ordinal is tree-automatic if and only if it is strictly below ω^{ω^ω}. Some complexity results for first-order theories of tree-automatic structures are shown in [17]. Recently, Huschenbett proved that it is decidable whether a given tree-automatic scattered linear order is string-automatic [9].

In this paper, we study tree-automatic well-founded trees.[1] Our main tool is a an ordinal measure for well-founded trees called ∞-rank, which is related to the classical (ordinal) rank of a well-founded tree. Consider a well-founded tree $\mathfrak{T}$ with root r. The rank of $\mathfrak{T}$ is the smallest ordinal, which is strictly larger than the ranks of the subtrees rooted in the children of r. In contrast to this, we only require the ∞-rank of $\mathfrak{T}$ to be (i) strictly larger than the ∞-ranks of those subtrees that are rooted (up to isomorphism) in *infinitely* many children of r and (ii) to be at least as large as the ∞-ranks of those subtrees that are rooted (up to isomorphism) in *finitely* many children of r. For instance, if a tree $\mathfrak{T}$ has finite depth, then $\infty\text{-rank}(\mathfrak{T})$ is the largest number $i \in \mathbb{N}$ such that the tree $\mathbb{N}^{\leq i}$ can be embedded into $\mathfrak{T}$.

Clearly, the ∞-rank of a well-founded tree is bounded from above by the classical (ordinal) rank of a tree. We also show that the rank of a well-founded tree of ∞-rank α is strictly bounded by $\omega \cdot (\alpha + 1)$. For string-automatic well-founded trees, it follows from [16] that the ∞-rank is always finite. Here, using a refinement of Delhommé's decomposition technique for tree-automatic structures [6], we show that the ∞-rank of a tree-automatic well-founded tree is strictly below ω^ω. As a corollary, we obtain that the rank of a string-automatic (resp., tree-automatic) well-founded tree is strictly below ω^2 (resp., ω^ω). The result for the string-automatic case nicely contrasts a result from [6,12], saying that the ranks of string-automatic well-founded partial orders reach exactly all ordinals below ω^ω.

Our second application of the ∞-rank concerns the isomorphism problem for tree-automatic well-founded trees. In [16], it was shown that the isomorphism problem for string-automatic well-founded trees is complete for level Δ^0_ω of the hyperarithmetical hierarchy. In other words, the isomorphism problem for string-automatic well-founded trees is recursively equivalent to true arithmetic. We show that the ∞-rank of well-founded computable trees determines the complexity of the isomorphism problem in the following sense: The isomorphism problem for well-founded computable trees of ∞-rank at most $\lambda + k$ (where $k \in \mathbb{N}$ and λ is a computable limit ordinal) belongs to level $\Sigma^0_{\lambda+3(k+1)}$ of the hyperarithmetical hierarchy. Since we know that the ∞-rank of a tree-automatic well-founded tree is strictly below ω^ω, we can use this fact and show that the isomorphism problem for tree-automatic well-founded trees belongs to level $\Delta^0_{\omega^\omega} = \Sigma^0_{\omega^\omega} \cap \Pi^0_{\omega^\omega}$ of the hyperarithmetical hierarchy. We also provide a corresponding lower bound w.r.t. Turing-reductions. Thus, the isomorphism problem for tree-automatic well-founded trees is $\Delta^0_{\omega^\omega}$-complete under Turing-reductions.

Let us remark that for non-well-founded order trees, the isomorphism problem is complete for Σ^1_1 (the first existential level of the analytical hierarchy) already in the string-automatic case [16], and this complexity is in a certain sense maximal, since the isomorphism problem for the class of all computable structures is Σ^1_1-complete as well

[1] In this paper *tree* always refers to an order tree $\mathfrak{T} = (T, \leq)$ as opposed to a successor tree, i.e., a tree is a partial order (without successor relation).

[5,7]. Let us also emphasize that all our results only hold for order trees, i.e., trees are seen as particular partial orders.

2 Preliminaries

A *relational structure* $\mathfrak{S}$ consists of a *domain* D and atomic relations on the set D. In this paper we will only consider structures with countable domains. Let $\mathfrak{A} = (A, \leq)$ be a partial order. A subset $B \subseteq A$ is a *chain* if for all $a, b \in B$, $a \leq b$ or $b \leq a$. A subset $B \subseteq A$ is an *antichain* if for all pairs of distinct $a, b \in B$, neither $a \leq b$ nor $b \leq a$.

Trees and forests. A *forest* is a partial order $\mathfrak{F} = (F, \leq)$ where for every $a \in F$ the set $\{b \in F \mid b < a\}$ is a finite chain. A *tree* is a forest which has a smallest element, which is called the *root* of the tree. Thus, a forest is a disjoint union of (an arbitrary number of) trees. For a given forest $\mathfrak{F}$, we denote by $\langle \mathfrak{F} \rangle$ the tree that results from adding a new root, i.e., a new smallest element, to $\mathfrak{F}$. If F is the domain of $\mathfrak{F}$ we denote by $\langle F \rangle$ the domain of $\langle \mathfrak{F} \rangle$. For a node u in $\mathfrak{F}$, let $\mathfrak{F}(u)$ be the subtree of $\mathfrak{F}$ at u, i.e., $\mathfrak{F}(u)$ is the restriction of $\mathfrak{F}$ to the set $\{v \in F \mid v \geq u\}$. We define the *successor relation* of $\mathfrak{F}$ as

$$E_{\mathfrak{F}} = \{(x, y) \in F \times F \mid x < y, \neg\exists z : x < z < y\}.$$

For $x \in F$ the set of *children* of x in $\mathfrak{F}$ is $E_{\mathfrak{F}}(x) = \{y \in F \mid (x, y) \in E_{\mathfrak{F}}\}$. A forest $\mathfrak{F} = (F, \leq)$ is *well-founded* if it does not contain an infinite ascending chain $a_1 < a_2 < a_3 < \cdots$.

Let us now define inductively the classical (ordinal) rank of a well-founded tree as well as the new notion of ∞-rank. We use standard terminology concerning ordinals; see e.g. [20]. For a set of ordinals M, let $\sup(M)$ be its supremum, where $\sup(\emptyset) = 0$. Let $\mathfrak{T}$ be a well-founded tree with root r. Thus, $C = E_{\mathfrak{T}}(r)$ is the set of children of the root. We define the *rank* of $\mathfrak{T}$ inductively as the ordinal

$$\mathsf{rank}(\mathfrak{T}) = \sup\{\mathsf{rank}(\mathfrak{T}(a)) + 1 \mid a \in C\}.$$

We define the ordinal $\infty\text{-}\mathsf{rank}(\mathfrak{T})$ inductively using $\alpha = \sup\{\infty\text{-}\mathsf{rank}(\mathfrak{T}(a)) \mid a \in C\}$:

$$\infty\text{-}\mathsf{rank}(\mathfrak{T}) = \begin{cases} \alpha & \text{if } \{a \in C \mid \infty\text{-}\mathsf{rank}(\mathfrak{T}(a)) = \alpha\} \text{ is finite,} \\ \alpha + 1 & \text{otherwise.} \end{cases}$$

The ∞-rank of a forest $\mathfrak{F}$ without smallest element is $\infty\text{-}\mathsf{rank}(\mathfrak{F}) = \infty\text{-}\mathsf{rank}(\langle \mathfrak{F} \rangle)$. A simple application of König's lemma shows that a well-founded forest $\mathfrak{F}$ has ∞-rank 0 if and only if $\mathfrak{F}$ is finite; in contrast, the rank of a finite tree can reach any finite ordinal. More generally, $\infty\text{-}\mathsf{rank}(\mathfrak{F}) = n < \omega$ if and only if there is an embedding of the tree $\mathbb{N}^{\leq n}$ (the tree of height n where every non-leaf has $\aleph_0$ many children) into $\langle \mathfrak{F} \rangle$ but no embedding of $\mathbb{N}^{\leq n+1}$ into $\langle \mathfrak{F} \rangle$. The following lemma is crucial for studying the ∞-rank:

Lemma 1. *Let $\mathfrak{F} = (F, \leq)$ be a well-founded forest. There are only finitely many $a \in F$ with $\infty\text{-}\mathsf{rank}(\mathfrak{F}(a)) = \infty\text{-}\mathsf{rank}(\mathfrak{F})$.*

Proof. Let $\alpha = \infty\text{-}\mathsf{rank}(\mathfrak{F})$. We show that $D = \{a \in \langle F \rangle \mid \infty\text{-}\mathsf{rank}(\langle\mathfrak{F}\rangle(a)) = \alpha\}$ is finite. Note that D is a downward-closed subset of the tree $\langle\mathfrak{F}\rangle$. Assume that this set is infinite. Since $\langle\mathfrak{F}\rangle$ is well-founded, König's lemma implies that D contains a node a which has infinitely many children a_i ($i \in \mathbb{N}$) that all belong to D. But then $\alpha = \infty\text{-}\mathsf{rank}(\langle\mathfrak{F}\rangle(a)) \geq \alpha + 1$, which is a contradiction. □

It is obvious that $\infty\text{-}\mathsf{rank}(\mathfrak{T}) \leq \mathsf{rank}(\mathfrak{T})$ for every well-founded tree $\mathfrak{T}$. On the other hand, we can also bound $\mathsf{rank}(\mathfrak{T})$ in terms of $\infty\text{-}\mathsf{rank}(\mathfrak{T})$ as follows.

Lemma 2. *For a well-founded tree $\mathfrak{T}$ we have* $\mathsf{rank}(\mathfrak{T}) < \omega \cdot (\infty\text{-}\mathsf{rank}(\mathfrak{T}) + 1)$.

Proof. Let $\mathfrak{T} = (T, \leq)$ We proceed by induction on $\infty\text{-}\mathsf{rank}(\mathfrak{T})$. If T is finite, then $\infty\text{-}\mathsf{rank}(\mathfrak{T}) = 0$ and $\mathsf{rank}(\mathfrak{T}) \leq |T| < \omega$. Now assume that $\infty\text{-}\mathsf{rank}(\mathfrak{T}) = \alpha$ for some ordinal $\alpha > 0$ such that the theorem holds for all trees of ∞-rank strictly below α. By Lemma 1, $T_\alpha = \{a \in T \mid \infty\text{-}\mathsf{rank}(\mathfrak{T}(a)) = \alpha\}$ is a finite and downward-closed subset of T. Let $M_\alpha \subseteq T_\alpha$ be the set of $\leq$-maximal elements of T_α and consider a tree $\mathfrak{T}(a)$ for $a \in M_\alpha$. The definition of M_α implies the following. If $b \in T$ with $b > a$, then $\infty\text{-}\mathsf{rank}(\mathfrak{T}(b)) = \beta$ for some ordinal $\beta < \alpha$. By the induction hypothesis it follows that $\mathsf{rank}(\mathfrak{T}(b)) < \omega \cdot (\beta + 1) \leq \omega \cdot \alpha$. In particular, $\mathsf{rank}(\mathfrak{T}(b)) < \omega \cdot \alpha$ for all children b of a. Thus, $\mathsf{rank}(\mathfrak{T}(a)) \leq \omega \cdot \alpha$. Finally, since T_α is a finite set, we have

$$\mathsf{rank}(\mathfrak{T}) \leq \sup\{\mathsf{rank}(\mathfrak{T}(a)) \mid a \in M_\alpha\} + |T_\alpha| \leq \omega \cdot \alpha + |T_\alpha| < \omega \cdot (\alpha + 1). \quad \square$$

Note that the upper bound of $\omega \cdot (\infty\text{-}\mathsf{rank}(\mathfrak{T}) + 1) = \omega \cdot \infty\text{-}\mathsf{rank}(\mathfrak{T}) + \omega$ is optimal as for any $n < \omega$, the linear order of size n has ∞-rank 0 but rank $\omega \cdot 0 + n$.

Finite labeled trees. A *finite binary tree* is a prefix-closed finite subset $T \subseteq \{0,1\}^*$, i.e., $uv \in T$ implies $u \in T$. We denote the set of all finite binary trees by $\mathcal{T}_2^{\mathsf{fin}}$. Let $\preceq$ be the prefix relation on $\{0,1\}^*$. Clearly $(T, \preceq)$ is a tree in the above sense.

Let Σ be a finite alphabet. A finite *Σ-labeled* binary tree is a pair (T, λ), where $T \in \mathcal{T}_2^{\mathsf{fin}}$ and $\lambda : T \to \Sigma$ is a labeling function. By $\mathcal{T}_{2,\Sigma}^{\mathsf{fin}}$ we denote the set of all finite Σ-labeled binary trees. Elements of $\mathcal{T}_{2,\Sigma}^{\mathsf{fin}}$ are denoted by lower case letters $(s, t, \ldots)$.

Next, we define the *convolution* $t_1 \otimes \cdots \otimes t_n$ of $t_1, \ldots, t_n \in \mathcal{T}_{2,\Sigma}^{\mathsf{fin}}$ as follows: Let $t_i = (T_i, \lambda_i)$ where $\lambda_i : T_i \to \Sigma$ and $\diamond \notin \Sigma$. Let $T = \bigcup_{i=1}^n T_i$ and define $\lambda_i' : T \to \Sigma \cup \{\diamond\}$ by $\lambda_i'(u) = \lambda_i(u)$ for $u \in T_i$ and $\lambda_i'(u) = \diamond$ for $u \in T \setminus T_i$. Then $t_1 \otimes \cdots \otimes t_n$ is the finite $((\Sigma \cup \{\diamond\})^n \setminus \{\diamond\}^n)$-labeled tree (T, λ) where λ is defined by $\lambda(u) = (\lambda_1'(u), \ldots, \lambda_n'(u))$ for each $u \in T$.

Tree automata and tree-automatic structures. For $T \in \mathcal{T}_2^{\mathsf{fin}}$ let

$$\mathsf{cl}(T) = T \cup \{ui \mid u \in T, i \in \{0,1\}\}$$

be its closure, which is again prefix-closed. Let Σ be a finite alphabet. A *tree automaton over Σ* is a tuple $\mathcal{A} = (Q, \Delta, Q_I, Q_F)$, where Q is the finite set of states, $Q_I \subseteq Q$ is the set of initial states, $Q_F \subseteq Q$ is the set of final states, and $\Delta \subseteq (Q \setminus Q_F) \times \Sigma \times Q \times Q$ is the transition relation. Given $t = (T, \lambda) \in \mathcal{T}_{2,\Sigma}^{\mathsf{fin}}$, a *successful run* of $\mathcal{A}$ on t is a mapping $\rho : \mathsf{cl}(T) \to Q$ such that (i) $\rho(\varepsilon) \in Q_I$, (ii) $\rho(\mathsf{cl}(T) \setminus T) \subseteq Q_F$, and (iii) for every $d \in T$, $(\rho(d), \lambda(d), \rho(d0), \rho(d1)) \in \Delta$. By $L(\mathcal{A})$ we denote the set of all $t \in \mathcal{T}^{\mathsf{fin}}_{2,\Sigma}$ on which $\mathcal{A}$ has a successful run. A set $L \subseteq \mathcal{T}^{\mathsf{fin}}_{2,\Sigma}$ is called *regular* if there is a tree automaton $\mathcal{A}$ over Σ with $L = L(\mathcal{A})$.

An n-ary relation $R \subseteq (\mathcal{T}^{\mathsf{fin}}_{2,\Sigma})^n$ is called *tree-automatic* if there is a tree automaton $\mathcal{A}_R$ over $(\Sigma \cup \{\diamond\})^n \setminus \{\diamond\}^n$ such that $L(\mathcal{A}_R) = \{t_1 \otimes \cdots \otimes t_n \mid (t_1, \ldots, t_n) \in R\}$. A relational structure $\mathfrak{S}$ is called *tree-automatic* over Σ if its domain is a regular subset of $\mathcal{T}^{\mathsf{fin}}_{2,\Sigma}$ and each of its atomic relations is tree-automatic; any tuple $\mathbb{P}$ of automata that accepts the domain and the relations of $\mathfrak{S}$ is called a *tree-automatic presentation* of $\mathfrak{S}$. In this case, we write $\mathfrak{S}(\mathbb{P})$ for $\mathfrak{S}$. If a tree-automatic structure $\mathfrak{S}$ is isomorphic to a structure $\mathfrak{S}'$, then $\mathfrak{S}$ is called a *tree-automatic copy* of $\mathfrak{S}'$ and $\mathfrak{S}'$ is *tree-automatically presentable*. In this paper we sometimes abuse the terminology referring to $\mathfrak{S}'$ as simply tree-automatic and calling a tree-automatic presentation of $\mathfrak{S}$ also a tree-automatic presentation of $\mathfrak{S}'$. We also simplify our statements by saying "given/compute a tree-automatic structure $\mathfrak{S}$" for "given/compute a tree-automatic presentation $\mathbb{P}$ of a structure $\mathfrak{S}(\mathbb{P})$". The structures $(\mathbb{N}, +)$ and $(\mathbb{N}, \times)$ are examples of tree-automatic structures. We will make use of the following simple lemma.

Lemma 3. *For every tree-automatic structure there is an isomorphic tree-automatic structure $\mathfrak{S}$ over the alphabet $\{a\}$, i.e., the domain can be seen as a subset of $\mathcal{T}^{\mathsf{fin}}_2$.*

Consider $\mathsf{FO} + \exists^\infty + \exists^{n,m} + \exists^{\mathsf{chain}}$, i.e., first-order logic extended by the quantifiers $\exists^\infty$ (there exists infinitely many), $\exists^{n,m}$ (there exists finitely many and the exact number is congruent n modulo m, where $m, n \in \mathbb{N}$) and the chain-quantifier $\exists^{\mathsf{chain}}$ (if $\varphi(x, y)$ is some formula, then $\exists^{\mathsf{chain}} \varphi(x, y)$ asserts that $\varphi(x, y)$ defines a partial order that contains an infinite ascending chain). Results from [3,10,21] show that the $\mathsf{FO} + \exists^\infty + \exists^{n,m} + \exists^{\mathsf{chain}}$ theory of any tree-automatic structure $\mathfrak{S}$ is (uniformly) decidable. Note that the property of being a tree is expressible in $\mathsf{FO} + \exists^\infty$ and well-foundedness of a tree is expressible in $\mathsf{FO} + \exists^{\mathsf{chain}}$. Hence, we get:

Theorem 4. *It is decidable whether a given tree-automatic structure is a well-founded tree.*

Let $\mathcal{K}$ be a class of tree-automatic presentations. The *isomorphism problem* $\mathrm{Iso}(\mathcal{K})$ is the set of pairs $(\mathbb{P}_1, \mathbb{P}_2) \in \mathcal{K} \times \mathcal{K}$ of tree-automatic presentations with $\mathfrak{S}(\mathbb{P}_1) \cong \mathfrak{S}(\mathbb{P}_2)$. If $\mathcal{K}$ is the class of tree-automatic presentations for a class $\mathcal{C}$ of relational structures (e.g. trees), then we will briefly speak of the isomorphism problem for (tree-automatic members of) $\mathcal{C}$. The isomorphism problem for the class of all tree-automatic structures is complete for Σ^1_1, the first level of the analytical hierarchy; this holds already for (non well-founded) string-automatic trees [14,16].

Hyperarithmetical sets. We use standard terminology concerning recursion theory; see e.g. [19]. We use the definition of the hyperarithmetical hierarchy from Ash and Knight [1] (cf. [8]). We first define inductively a set of *ordinal notations* $O \subseteq \mathbb{N}$. Simultaneously we define a mapping $a \mapsto |a|_O$ from O into ordinals and a strict partial order $<_O$ on O. The set O is the smallest subset of $\mathbb{N}$ satisfying the following conditions:

- $1 \in O$ and $|1|_O = 0$, i.e., 1 is a notation for the ordinal 0.

– If $a \in O$, then also $2^a \in O$. We set $|2^a|_O = |a|_O + 1$ and let $b <_O 2^a$ if and only if $b = a$ or $b <_O a$.
– If $e \in \mathbb{N}$ is such that Φ_e (the e^{th} partial computable function) is total, $\Phi_e(n) \in O$ for all $n \in \mathbb{N}$, and $\Phi_e(0) <_O \Phi_e(1)_O <_O \Phi_e(2) <_O \cdots$, then also $3 \cdot 5^e \in O$. We set $|3 \cdot 5^e|_O = \sup\{|\Phi_e(n)|_O \mid n \in \mathbb{N}\}$ and let $b <_O 3 \cdot 5^e$ if and only if there is an $n \in \mathbb{N}$ with $b <_O \Phi_e(n)$.

An ordinal α is *computable* if there is an $a \in O$ with $|a|_O = \alpha$. The smallest non-computable ordinal is the Church-Kleene ordinal ω_1^{ck}. If $a \in O$ then the restriction of the partial order $(O, <_O)$ to $O_a = \{b \in O \mid b <_O a\}$ is isomorphic to the ordinal $|a|_O$ [1, Proposition 4.9]. Based on ordinal notations we define the *hyperarithmetical hierarchy*. For this we define sets $H(a)$ for each $a \in O$ as follows:

– $H(1) = \emptyset$,
– $H(2^b) = H(b)'$ (the Turing jump of $H(b)$; see e.g. [19]),
– $H(3 \cdot 5^e) = \{\langle b, n\rangle \mid b <_O 3 \cdot 5^e, n \in H(b)\}$; here $\langle \cdot, \cdot \rangle$ denotes some computable pairing function.

Spector has shown that $|a|_O = |b|_O$ implies that $H(a)$ and $H(b)$ are Turing equivalent. The levels of the hyperarithmetical hierarchy can be defined as follows, where α is a computable ordinal.

– Σ^0_α is the set of all subsets $A \subseteq \mathbb{N}$ that are recursively enumerable in some $H(a)$ with $|a|_O = \alpha$ (by Spector's theorem, the concrete choice of a is irrelevant).
– Π^0_α is the set of all complements of Σ^0_α sets.
– $\Delta^0_\alpha = \Sigma^0_\alpha \cap \Pi^0_\alpha$, i.e., Δ^0_α is the set of all subsets $A \subseteq \mathbb{N}$ that are Turing-reducible to some $H(a)$ with $|a|_O = \alpha$.

For any two computable ordinals α and β, $\alpha < \beta$ implies $\Sigma_\alpha \cup \Pi_\alpha \subsetneq \Delta_\beta$. The union of all classes Σ^0_α where $\alpha < \omega_1^{\mathsf{ck}}$ yields the class of all *hyperarithmetical sets*. By a classical result of Kleene, the hyperarithmetical sets are exactly the sets in $\Delta^1_1 = \Sigma^1_1 \cap \Pi^1_1$, where Σ^1_1 is the first existential level of the analytical hierarchy, and Π^1_1 is the set of all complements of Σ^1_1-sets.

3 Bounding the ∞-rank of tree-automatic well-founded trees

The first main result of this paper is:

Theorem 5. *If $\mathfrak{T}$ is a tree-automatic well-founded tree, then* $\infty\text{-}\mathsf{rank}(\mathfrak{T}) < \omega^\omega$.

Before we sketch a proof of this result, let us first deduce a corollary:

Corollary 6. *For $\mathfrak{T} = (T, \leq)$ a string-automatic (tree-automatic, respectively) well-founded tree we have* $\mathsf{rank}(\mathfrak{T}) < \omega^2$ *(*$\mathsf{rank}(\mathfrak{T}) < \omega^\omega$*, respectively).*

Proof. For a string-automatic well-founded tree $\mathfrak{T}$, $\infty\text{-}\mathsf{rank}(\mathfrak{T})$ is finite by [16][2]. With Lemma 2 we get $\mathsf{rank}(\mathfrak{T}) \leq \omega \cdot i < \omega^2$ for some $i \in \mathbb{N}$. For a tree-automatic well-founded tree $\mathfrak{T}$ we have $\infty\text{-}\mathsf{rank}(\mathfrak{T}) < \omega^\omega$ by Theorem 5. Thus, there is some $i \in \mathbb{N}$ such that $\infty\text{-}\mathsf{rank}(\mathfrak{T}) \leq \omega^i$. With Lemma 2 we get $\mathsf{rank}(\mathfrak{T}) < \omega \cdot \omega^i + \omega < \omega^\omega$. □

[2] In [16], the notion of *embedding rank* of an arbitrary tree is defined. Comparison of the definitions shows that the embedding rank of a well-founded tree is finite iff its ∞-rank is finite.

Note that Corollary 6 contrasts with results on the ranks of string-automatic well-founded partial orders.[3] By [6,12], the ordinal ranks of string-automatic well-founded partial orders are the ordinals strictly below ω^ω. In fact, the result still holds for partial orders without infinite chains [11]. Moreover, Delhommé's characterization of tree-automatic ordinals yields tree-automatic well-founded partial orders of rank α for each $\alpha < \omega^{\omega^\omega}$ [6].

Let us sketch the proof of Theorem 5. The first part relies on Delhommé's decomposition technique for tree-automatic structures from [6] where he proved that the ordinal ω^{ω^ω} is not tree-automatic. Let us explain his decomposition technique for a tree-automatic graph $\mathfrak{G} = (V, E)$. Because of Lemma 3 we can assume that $V \subseteq \mathcal{T}_2^{\mathsf{fin}}$. Consider a tree automaton $\mathcal{A}$ that accepts a subset of $V \otimes \mathcal{T}_2^{\mathsf{fin}}$, and for each $s \in \mathcal{T}_2^{\mathsf{fin}}$ let $\mathfrak{G}_s$ be the subgraph of $\mathfrak{G}$ induced by the set $\{t \in V \mid t \otimes s \in L(\mathcal{A})\}$.

Delhommé's main proposition from [6] shows that every subgraph $\mathfrak{G}_s$ can be obtained from a finite set of subgraphs $\mathcal{C}$ by using the operations of *box-augmentation* and *sum-augmentation*. Roughly speaking, a graph $\mathfrak{G}$ is a sum-augmentation of subgraphs $\mathfrak{G}_1, \ldots, \mathfrak{G}_n$ if it is the disjoint union of $\mathfrak{G}_1, \ldots, \mathfrak{G}_n$ where we may add edges between different $\mathfrak{G}_i$ (but not within a single $\mathfrak{G}_i$). $\mathfrak{G}$ is a box-augmentation of the graphs $\mathfrak{G}_1, \ldots, \mathfrak{G}_n$ with node sets $V_1, \ldots, V_n$ if the node set of $\mathfrak{G}$ is the product $\prod_{i=1}^n V_i$ and for every $1 \leq i \leq n$ and all $v_1 \in V_1, \ldots, v_{i-1} \in V_{i-1}, v_{i+1} \in V_{i+1}, \ldots, v_n \in V_n$, the subgraph of $\mathfrak{G}$ induced by the set $\{(v_1, \ldots, v_{i-1}, v, v_{i+1}, \ldots, v_n) \mid v \in V_i\}$ is isomorphic to $\mathfrak{G}_i$.

Now, let ν be a function that maps graphs to some set M such that isomorphic graphs are mapped to the same element. We say that $m \in M$ is ν-sum-indecomposable (ν-box-indecomposable, resp.) if for all graphs $\mathfrak{G}, \mathfrak{G}_1, \ldots, \mathfrak{G}_n$ such that $\mathfrak{G}$ is a sum-augmentation (box-augmentation, resp.) of $\mathfrak{G}_1, \ldots, \mathfrak{G}_n$ the following implication holds: If $\nu(\mathfrak{G}) = m$ then $\nu(\mathfrak{G}_i) = m$ for some $1 \leq i \leq n$. Delhommé's decomposition result implies that the set $\{\nu(\mathfrak{G}_s) \mid s \in \mathcal{T}_2^{\mathsf{fin}}\}$ contains only finitely many values that are both ν-sum-indecomposable and ν-box-indecomposable.

In order to show that ω^{ω^ω} is not tree-automatic, Delhommé takes a tree-automatic copy $\mathfrak{G} = (V, \leq)$ of some ordinal and a tree automaton $\mathcal{A}$ for the first-order formula $y < x$. Hence, the substructures $\mathfrak{G}_s$ are the initial segments of $\mathfrak{G}$. Moreover let ν_0 be the function that maps an initial segment of $\mathfrak{G}$ to the corresponding ordinal. Delhommé proves that every ordinal of the form ω^{ω^α} is both ν_0-sum-indecomposable and ν_0-box-indecomposable. Hence, $\mathfrak{G} = (V, \leq)$ can contain only finitely many initial segments of the form ω^{ω^α}, which is not the case for ω^{ω^ω}.

We follow a similar strategy. Heading for a contradiction to Theorem 5, take a well-founded tree-automatic forest $\mathfrak{F} = (F, \leq)$ with $\infty\text{-}\mathsf{rank}(\mathfrak{F}) = \omega^\omega$. Let $\mathcal{A}$ be a tree automaton for the first-order formula $x \leq y$. Hence, the substructures $\mathfrak{F}_s$ are the subtrees $\mathfrak{F}(v)$ of $\mathfrak{F}$ for $v \in F$. It is not difficult to show that for every ordinal $\alpha < \omega^\omega$, $\mathfrak{F}$ must contain a subtree of ∞-rank α. In particular, $\mathfrak{F}$ contains a subtree of ∞-rank ω^i for every $i \in \mathbb{N}$. Now, let ν_1 be the function that maps a subtree $\mathfrak{F}(v)$ to its ∞-rank. We would obtain a contradiction by proving that every ordinal of the form ω^α is both ν_1-sum-indecomposable and ν_1-box-indecomposable. Indeed, we can prove that every

[3] The rank generalizes naturally to all well-founded partial orders (considering roots as maximal elements of trees).

ordinal of the form ω^α is ν_1-sum-indecomposable. But there is a problem with ν_1-box-indecomposability: The ordinals 0 and 1 are the only ν_1-box-indecomposable ordinals. The problem is that any forest can be embedded into the box-augmentation of two infinite antichains. Hence, box-augmentations of two infinite antichains may have arbitrarily high ∞-rank. To solve this problem, we observe that the box-augmentations that are used for building up the subtrees $\mathfrak{F}_s$ of $\mathfrak{F}$ ($s \in \mathcal{T}_2^{\mathsf{fin}}$) have a particular property that we call tame colorability (this is joint work with Martin Huschenbett). If a graph $\mathfrak{G} = (V, E)$ is a box-augmentation of subgraphs $\mathfrak{G}_i = (V_i, E_i)$ ($1 \le i \le n$), then this box-augmentation is *tamely colorable* if for each $1 \le i \le n$ there is a finite coloring c_i of $V_i \times V_i$ such that whether $((v_1, \dots v_n), (v'_1, \dots, v'_n)) \in E$ only depends on the colors $c_i(v_i, v'_i)$ for $1 \le i \le n$. A careful analysis of box-augmentations of forests shows that ω^α is also ν_1-tamely-colorable-box-indecomposable (tamely-colorable-box-indecomposability is defined as box-indecomposability, but only considering tamely colorable box-augmentations). As in Delhommé's argument, we conclude that a well-founded tree-automatic forest $\mathfrak{F}$ only contains finitely many subtrees of pairwise distinct ∞-ranks of the form ω^i. Hence, $\infty\text{-rank}(\mathfrak{F}) < \omega^\omega$ and Theorem 5 follows.

4 The isomorphism problem for well-founded tree-automatic trees

It turns out that the ∞-rank for well-founded computable trees yields an upper bound on the recursion-theoretic complexity of the isomorphism problem. Recall that we defined trees and forests as particular partial orders. For the isomorphism problem, it is useful to assume that also the direct successor relation is computable. When speaking of a *computable forest* in the following theorem, we mean a forest $\mathfrak{F} = (F, \le)$ such that $F \subseteq \mathbb{N}$, $\le\, \subseteq \mathbb{N} \times \mathbb{N}$, and the direct successor relation $E_{\mathfrak{F}}$ are all computable sets.[4] Note that the direct successor relation of a tree-automatic forest is still tree-automatic (and hence computable) because it is first-order definable.

Lemma 7. *Let α be a computable ordinal and assume that $\alpha = \lambda + k$, where $k \in \mathbb{N}$ and either $\lambda = 0$ or λ is a limit ordinal. The isomorphism problem for well-founded computable trees of ∞-rank at most α belongs to level $\Sigma^0_{\lambda+2(k+1)}$ of the hyperarithmetical hierarchy.*

For the proof of Lemma 7 we use a characterization of the hyperarithmetical levels by *computable infinitary formulas* (cf. [1]). These are first-order formulas over the structure $(\mathbb{N}, +, \times)$, where countably infinite conjunctions and disjunctions are allowed. Computability of such an infinitary formula means that for an infinite conjunction $\bigvee_{n \in \mathbb{N}} \varphi_n$ there is a computable function that maps n to a representation of φ_n (note that φ_n may again contain infinite conjunctions and disjunctions), and similarly for infinite disjunctions. Roughly speaking, such an infinitary formula can be encoded by a computable tree (the syntax tree of the formula) and if that tree has rank α (a computable ordinal) then the relation defined by the formula belongs to level α of the hyperarithmetical hierarchy.

[4] On the other hand, if we would omit the requirement of a computable direct successor relation in Lemma 7, then we would only have to replace the constant 2 in the lemma by a larger value.

In order to prove Lemma 7, we construct for a computable ordinal α a computable infinitary formula $\mathsf{iso}_\alpha(x,y)$ that is satisfied in a well-founded computable forest $\mathfrak{F}$ if and only if $\mathfrak{F}(x)$ and $\mathfrak{F}(y)$ have ∞-rank at most α and $\mathfrak{F}(x) \cong \mathfrak{F}(y)$. The construction is carried out inductively along the ordinal α, similarly to the proof of Lemma 25 in [16].

From Theorem 5 and Lemma 7 it follows that the isomorphism problem for well-founded tree-automatic trees belongs to $\Pi^0_{\omega^\omega}$. Using similar formulas as those constructed in our proof of Lemma 7, we can also show that the isomorphism problem for well-founded tree-automatic trees belongs to $\Sigma^0_{\omega^\omega}$. Hence, we get:

Corollary 8. *The isomorphism problem for well-founded tree-automatic trees belongs to* $\Delta^0_{\omega^\omega}$.

Let us now turn to lower bounds. Our main technical result is:

Lemma 9. *From a given $i \in \mathbb{N}$, one can compute a well-founded tree-automatic tree $\mathfrak{V}_i$ such that the following holds: From a given $\Pi^0_{\omega^i}$-set $P \subseteq \mathbb{N}$ (represented e.g. by a computable infinitary formula) and $n \in \mathbb{N}$ one can compute a well-founded tree-automatic tree $\mathfrak{W}_{P,n}$ such that $n \in P$ if and only if $\mathfrak{V}_i \cong \mathfrak{W}_{P,n}$.*

We prove Lemma 9 by a reduction from the isomorphism problem for well-founded *computable trees*. We use a construction from [8]. Basically, [8, Proposition 3.2] states Lemma 9 for well-founded computable trees (instead of well-founded tree-automatic trees) and all computable ordinals (instead of ordinals ω^i). It turns out that the trees constructed in [8] for a certain ordinal α consist of computable subtrees of a "universal" well-founded computable tree $\mathfrak{S}_\alpha$. In case $\alpha = \omega^i$ for $i \in \mathbb{N}$ we can moreover show that the tree $\mathfrak{S}_{\omega^i}$ is tree-automatic. Roughly speaking this yields a weaker version of Lemma 9, where instead of well-founded tree-automatic trees we have the (tree-automatic) trees $\mathfrak{S}_{\omega^i}$ enriched by a computable unary predicate K on the node set of $\mathfrak{S}_{\omega^i}$. In fact K can be assumed to be a subset of the leaves of $\mathfrak{S}_{\omega^i}$; it yields a computable subtree of the universal tree $\mathfrak{S}_{\omega^i}$ by removing all leaves from K. Finally, to get rid of K we use a technique from [16]. In Lemma 41 from [16] it was shown that there are non-isomorphic (string-automatic) trees U_0 and U_1 with the following property: from an index of a computable set of strings $L \subseteq \{0,1\}^*$ one can compute a string-automatic forest $\mathfrak{F}_L$ of height 3 such that: (i) the set of roots is $\{0,1\}^*$, (ii) if $x \in L$ then $\mathfrak{F}_L(x) \cong U_0$, and (iii) if $x \notin L$ then $\mathfrak{F}_L(x) \cong U_1$. In [16], this statement was used in order to reduce the isomorphism problem for (non-well-founded) computable trees to the isomorphism problem for (non-well-founded) string-automatic trees. Hence, the latter problem is Σ^1_1-complete. In our situation, we first prove a tree version of [16, Lemma 41], where $\{0,1\}^*$ is replaced by $\mathcal{T}_2^{\mathsf{fin}}$. Then, one can eliminate the additional computable unary predicate K on the leaves of $\mathfrak{S}_{\omega^i}$. For this, every leaf from K is replaced by the height-3 tree U_0, whereas all other leaves are replaced by the height-3 tree U_1. This yields a well-founded tree automatic tree that encodes the pair $(\mathfrak{S}_{\omega^i}, K)$.

From Lemma 9 we can deduce that the isomorphism problem for well-founded tree-automatic trees is $\Delta^0_{\omega^\omega}$-hard under Turing-reductions. With Corollary 8, we finally obtain our main result for the isomorphism problem.

Theorem 10. *The isomorphism problem for well-founded tree-automatic trees is $\Delta^0_{\omega^\omega}$-complete under Turing-reductions.*

Acknowledgment We thank Martin Huschenbett for helpful discussions concerning Delhommé's decomposition result.

References

1. C. J. Ash and J. F. Knight. *Computable structures and the hyperarithmetical hierarchy*, volume 144 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, 2000.
2. V. Bárány, E. Grädel, and S. Rubin. Automata-based presentations of infinite structures. In *Finite and Algorithmic Model Theory*, number 379 in London Mathematical Society Lecture Notes Series, pages 1–76. Cambridge University Press, 2011.
3. A. Blumensath. Automatic structures. Diploma thesis, RWTH Aachen, 1999.
4. A. Blumensath and E. Grädel. Finite presentations of infinite structures: Automata and interpretations. *Theory Comput. Syst.*, 37:642–674, 2004.
5. W. Calvert and J. F. Knight. Classification from a computable viewpoint. *Bull. Symbolic Logic*, 12(2):191–218, 2006.
6. C. Delhommé. Automaticité des ordinaux et des graphes homogènes. *C.R. Acad. Sci. Paris Ser. I*, 339:5–10, 2004.
7. S. S. Goncharov and J. F. Knight. Computable structure and antistructure theorems. *Algebra Logika*, 41(6):639–681, 2002.
8. D. R. Hirschfeldt and W. M. White. Realizing levels of the hyperarithmetic hierarchy as degree spectra of relations on computable structures. *Notre Dame J. Form. Log.*, 43(1):51–64, 2002.
9. M. Huschenbett. Word automaticity of tree automatic scattered linear orderings is decidable. Technical report, arXiv.org, 2012. `http://arxiv.org/abs/1201.5070`.
10. A. Kartzow. *First-Order Model Checking On Generalisations of Pushdown Graphs*. PhD thesis, TU Darmstadt, 2011.
11. A. Kartzow, M. Lohrey and J. Liu. Tree-automatic well-founded trees. Technical report, arXiv.org, 2012. `http://arxiv.org/abs/1201.5495`.
12. B. Khoussainov and M. Minnes. Model theoretic complexity of automatic structures. *Ann. Pure Appl. Logic*, 161(3):416–426, 2009.
13. B. Khoussainov and A. Nerode. Automatic presentations of structures. In *LCC: International Workshop on Logic and Computational Complexity*, LNCS 960, pages 367–392. Springer, 1995.
14. B. Khoussainov, A. Nies, S. Rubin, and F. Stephan. Automatic structures: richness and limitations. *Log. Methods Comput. Sci.*, 3(2):2:2, 18 pp. (electronic), 2007.
15. B. Khoussainov, S. Rubin, and F. Stephan. Automatic linear orders and trees. *ACM Trans. Comput. Log.*, 6(4):675–700, 2005.
16. D. Kuske, J. Liu, and M. Lohrey. The isomorphism problem on classes of automatic structures with transitive relations. to appear in *Trans. Amer. Math. Soc.*, 2012.
17. D. Kuske and M. Lohrey. Automatic structures of bounded degree revisited. *J. Symbolic Logic*, 76(4):1352–1380, 2011.
18. G. P. Oliver and R. M. Thomas. Automatic presentations for finitely generated groups. In *Proc. STACS 2005*, LNCS 3404, pages 693–704. Springer, 2005.
19. H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, 1968.
20. J. Rosenstein. *Linear Ordering*. Academic Press, 1982.
21. A. W. To and L. Libkin. Recurrent reachability analysis in regular model checking. In *Proc. LPAR 2008*, LNCS 5330, pages 198–213, 2008.