On Automatic Partial Orders

Bakhadyr Khoussainov
Department of Computer Science
University of Auckland, New Zealand
bmk@cs.auckland.ac.nz

Sasha Rubin
Department of Mathematics
University of Auckland, New Zealand
rubin@math.auckland.ac.nz

Frank Stephan
Mathematical Institute
University of Heidelberg, Germany
fstephan@math.uni-heidelberg.de

Abstract

We investigate partial orders that are computable, in a precise sense, by finite automata. Our emphasis is on trees and linear orders. We study the relationship between automatic linear orders and trees in terms of rank functions that are versions of Cantor-Bendixson rank. We prove that automatic linear orders and automatic trees have finite rank. As an application we provide a procedure for deciding the isomorphism problem for automatic ordinals. We also investigate the complexity and definability of infinite paths in automatic trees. In particular, we show that every infinite path in an automatic tree with countably many infinite paths is a regular language.

1. Introduction

Consider a class of infinite structures, such as the class of graphs, partial orders, trees, groups, or lattices, etc. A given structure in this class may or may not be computable. If it is one then naturally asks whether or not the structure, or algorithmic problems of the structure, are feasibly computable. The area of automatic structures studies those structures that are computable, in a certain precise sense, by finite automata. The automata in this paper are synchronous automata operating on finite words. Using the closure of these automata under boolean operations and projection, we get that the first order theory of an automatic structure is decidable (see, e.g. [10]). This is a result that motivates study of automatic structures in computer science. For example, a related concept to the one presented here is that of automatic groups from computational group theory [7]. There they prove that a finitely generated automatic group is finitely presentable and that its word problem is solvable in quadratic time. The general notion of structures presentable by automata has been recently studied in [1],[2],[5],[8],[10],[12]. Throughout this paper we will use the following more general theorem proved by Grädel and Blumensath (see [2]) without explicit mention.

Theorem 1.1 *Given an automatic structure $\mathcal{A}$ and a relation R which is first order definable (with the quantifier $\exists^\infty$ which stands for "there exist infinitely many") in $\mathcal{A}$, one can effectively construct an automaton recognising R.*

This paper studies automatic partial orderings with an emphasis on trees and linear orderings. A **partial order (partial ordering)** is a pair $(A, \preceq)$ such that $\preceq$ is a reflexive, transitive and anti-symmetric binary relation on the nonempty domain A. A **linear order** $\mathcal{L}$ is a partial order $(L, \leq)$ in which $\leq$ is total, that is $\forall x \forall y (x \leq y \vee y \leq x)$. A general problem of our interest concerns characterising the isomorphism types of the automatic linear orders. Classically, linear orderings are characterised in terms of scattered and dense linear orderings as follows. We say that $\mathcal{L}$ is **dense** if for all distinct a and b in L with $a < b$ there exists an $x \in L$ with $a < x < b$. There are only five types of countable dense linear orderings up to isomorphism: the order of rational numbers with or without least and greatest elements, and the order type of the trivial linear order with exactly one element. We say that $\mathcal{L}$ is **scattered** if it does not contain a nontrivial dense subordering. Examples, of scattered linear orders are finite sums of cartesian products of order types of ω and Z (integers).

Theorem 1.2 see [14, Theorem 4.9] *Every countable linear ordering $\mathcal{L}$ can be represented as a dense sum of countable scattered linear orderings.*

The scattered linear orderings can be characterised inductively, whereby to each linear order $\mathcal{L}$ one associates a

Proceedings of the 18th Annual IEEE Symposium on Logic in Computer Science (LICS'03)
1043-6871/03 $17.00 © 2003 IEEE

countable ordinal – called the FC–rank of $\mathcal{L}$, a version of Cantor-Bendixson rank for topological spaces. One of our results in this paper gives an upper bound on the FC–rank of automatic linear orders. For scattered linear orders, the FC–rank coincides with the VD–rank (these are defined in the next section).

Theorem 3.5 *If $\mathcal{L}$ is an automatic linear order, then its FC–rank is finite.*

The proof of this theorem generalises a novel technique of Delhomme who gives a full characterisation of automatic ordinals.

Corollary 1.3 [4] *An ordinal α is automatic if and only if $\alpha < \omega^\omega$.*

Another class of structures discussed in this paper are trees. A **tree** $\mathcal{T} = (T, \preceq)$ is a partial order that has a minimum element and in which every set of the form $\{y \mid y \prec x\}$ forms a finite linear order. Elements of trees are called nodes. A node $y \in T$ is an immediate successor of $x \in T$ if $x \prec y$ and there does not exist z for which $x \prec z \prec y$. A tree $\mathcal{T}$ is **finitely branching** if each node $x \in T$ has only finitely many immediate successors. A **path** of a tree $(T, \preceq)$ is a subset $P \subseteq T$ which is linearly ordered, closed downward (that is, whenever $y \in P$ and $x \preceq y$ then $x \in P$) and maximal (with respect to set theoretic inclusion) with these properties. An **infinite path** is a path P where $|P|$ is infinite. We are interested in understanding algebraic, model-theoretic as well as computational properties of automatic trees.

In analogy to Kleene-Brouwer orderings in which trees are associated with linear orderings (see [13]), we build linear orderings from trees. This transformation preserves automaticity, and associates the Cantor-Bendixson rank (CB-rank for short) of trees with the VD-ranks of linear orders obtained. Informally, the CB-rank of the tree tells us how big the tree is in terms of ordinals (see [9] for example). This relationship between trees and linear orders gives us the next result:

Theorem 5.5 *The CB-rank of an automatic tree with countably many infinite paths is finite.*

It is known that every infinite finitely branching tree has an infinite path (König's Lemma). The proof of this fact does not produce an infinite path constructively. In fact, there are even examples of computable finitely branching trees with *exactly* one infinite path, and the path is *not* computable. Moreover, if one omits the assumption that the tree is finitely branching then there are examples of computable trees in which every infinite path is not even arithmetical (see [13]). This negative phenomenon fails dramatically when one considers automatic trees, and not only finitely branching ones.

Theorem 4.1 *If an automatic tree has an infinite path, then it has a regular infinite path.*

We can significantly strengthen this theorem under the assumption that the tree has at most countably many paths. Indeed, from Theorem 5.4 one can derive that if an automatic finitely branching tree $\mathcal{T}$ has countably many infinite paths then every path of $\mathcal{T}$ is regular. This is because the set of paths in such trees is definable. However, we can even omit the assumption that the tree is finitely branching:

Theorem 4.4 *If an automatic tree has countably many infinite paths then every infinite path in it is regular.*

All classical definitions and unproved results on linear orderings can be found in Rosenstein [14]. Countable means finite or countably infinite. All structures are assumed to be countable. Definable means first order definable with the additional quantifier $\exists^\infty$.

2. Preliminaries

A thorough introduction to automatic structures can be found in [1] and [10]. A recent survey paper [11] discusses the basic results and possible directions for future work in the area. Familiarity with the basics of finite automata theory is assumed though for completeness the necessary definitions are included here.

A *finite automaton* $\mathcal{A}$ over an alphabet Σ is a tuple (S, ι, Δ, F), where S is a finite set of *states*, $\iota \in S$ is the *initial state*, $\Delta \subset S \times \Sigma \times S$ is the *transition table* and $F \subset S$ is the set of *final states*. A *computation* of $\mathcal{A}$ on a word $\sigma_1\sigma_2 \ldots \sigma_n$ ($\sigma_i \in \Sigma$) is a sequence of states, say $q_0, q_1, \ldots, q_n$, such that $q_0 = \iota$ and $(q_i, \sigma_{i+1}, q_{i+1}) \in \Delta$ for all $i \in \{0, 1, \ldots, n-1\}$. If $q_n \in F$, then the computation is *successful* and we say that automaton $\mathcal{A}$ *accepts* the word. The *language* accepted by the automaton $\mathcal{A}$ is the set of all words accepted by $\mathcal{A}$. In general, $D \subset \Sigma^\star$ is *finite automaton recognisable*, or *regular*, if D is equal to the language accepted by $\mathcal{A}$ for some finite automaton $\mathcal{A}$.

Classically finite automata recognise sets of words. The following definitions extends recognisability to relations of arity n, called *synchronous n–tape automata*. Informally a synchronous n–tape automaton can be thought of as a one-way Turing machine with n input tapes [6]. Each tape is regarded as semi-infinite having written on it a word in the alphabet Σ followed by an infinite succession of blanks, $\diamond$ symbols. The automaton starts in the initial state, reads simultaneously the first symbol of each tape, changes state, reads simultaneously the second symbol of each tape, changes state, etc., until it reads a blank on each tape. The automaton then stops and accepts the n–tuple of words if it is in a final state. The set of all n–tuples accepted by the automaton is the relation recognised by the automaton. Here is a formalisation:

Definition 2.1 *Let $\Sigma_\diamond$ be $\Sigma \cup \{\diamond\}$ where $\diamond \notin \Sigma$. The* convolution of a tuple $(w_1, \cdots, w_n) \in (\Sigma^*)^n$ *is the tuple* $(w_1, \cdots, w_n)^\diamond \in ((\Sigma_\diamond)^*)^n$ *formed by concatenating the least number of blank symbols, $\diamond$, to the right ends of the w_i, $1 \leq i \leq n$, so that the resulting words have equal length. The* convolution of a relation $R \subset (\Sigma^*)^n$ *is the relation $R^\diamond \subset ((\Sigma_\diamond)^*)^n$ formed as the set of convolutions of all the tuples in R.*

Definition 2.2 An n–tape automaton *on Σ is a finite automaton over the alphabet $(\Sigma_\diamond)^n$. An n–ary relation $R \subset \Sigma^{\star n}$ is* finite automaton recognisable *or* regular *if its convolution $R^\diamond$ is recognisable by an n–tape automaton.*

We now relate n–tape automata to structures. A *structure* $\mathcal{A}$ consists of a set A called the *domain* and some constants, relations and operations on A. We may assume that $\mathcal{A}$ only contains relational and constant predicates as the operations can be replaced with their graphs. We write $\mathcal{A} = (A, R_1^A, \ldots, R_k^A, c_0^A, \ldots, c_t^A)$ where R_i^A is an n_i–ary relation on $\mathcal{A}$ and c_j^A is a constant element of $\mathcal{A}$.

Definition 2.3 *A structure $\mathcal{A}$ is* automatic *over Σ if its domain $A \subset \Sigma^\star$ and the relations $R_i^A \subset \Sigma^{*n_i}$ all are finite automaton recognisable.*

An isomorphism from a structure $\mathcal{B}$ to an automatic structure $\mathcal{A}$ is an automatic presentation *of $\mathcal{B}$ in which case $\mathcal{B}$ is called* automatically presentable *(over Σ). A structure is called* automatic *if it is automatic over some alphabet.*

Examples of automatic or automatically presentable structures are Presburger arithmetic $(\omega, S, +, 0)$, the group of integers $(Z, +)$, the Boolean algebra of finite or co-finite subsets of ω, the word structure $(\{0,1\}^\star, L, R, E, \preceq)$, where for all strings $x, y \in \{0,1\}^\star$ we have $L(x) = x0$, $R(x) = x1$, $E(x, y)$ iff $|x| = |y|$, and $\preceq$ is the lexicographical order. Examples of automatic or automatically presentable linear orders are $(\omega, \leq)$, $(Z, \leq)$ and the order on rationals $(Q, \leq)$. Moreover, if $\mathcal{L}_1 = (L_1, \leq_1)$ and $\mathcal{L}_2 = (L_2, \leq_2)$ are automatic linear orders then so are their sum and product. Below we present two examples that generate automatic trees.

Example 2.4 *Let R be a regular language. Consider the partial order $\mathcal{T} = (Pref(R), \leq)$, where $Pref(R)$ is the language of all prefixes of strings from R, and $\leq$ is the prefix relation. Then $\mathcal{T}$ is an automatic tree.*

Example 2.5 *Let R be a regular language. Consider the partial order $\mathcal{T} = (R, \leq)$, where $x \leq y$ iff $x = y$ or $|x| < |y|$ and x is lexicographically smallest among all $x' \in R$ such that $|x| = |x'|$. Then $\mathcal{T}$ is an automatic tree.*

We now need some facts and notations about linear orders. Write ω for the type of the positive integers, $\omega^\star$ for the negative integers, ζ for the integers, η for the rationals and $\mathbf{n}$ for the finite order on n elements. A **closed interval** written $[x, y]$ is $\{z \mid x \leq z \leq y\}$ if $x \leq y$ and $\{z \mid y \leq z \leq x\}$ otherwise. If $\mathcal{L}$ is a linear ordering, then unless specified we denote its domain by L and ordering by $\leq_L$ or simply $\leq$.

Definition 2.6 *Consider a linear order I as an index set for a set of linear orderings $\{\mathcal{A}_i\}_{i \in I}$. The I*-**sum**

$$\mathcal{L} = \Sigma\{\mathcal{A}_i \mid i \in I\}$$

is the linear order with domain $\cup_i A_i$. For $x \in A_i, y \in A_j$ define $x \leq_L y$ if $(i <_I j) \vee (i = j \wedge x \leq_{A_i} y)$.

We refer to the case when I is dense as a **dense sum**. We need the following definition of VD-rank and the class VD (very discrete) of scattered linear orders.

Definition 2.7 *For each countable ordinal α, define the set VD_α of linear orders inductively as*

1. *$VD_0 := \{\mathbf{0}, \mathbf{1}\}$, where $\mathbf{0}$ is the empty ordering and $\mathbf{1}$ is the ordering with exactly one element.*

2. *$VD_\alpha :=$ the set of linear orderings formed as I–sums where the $\{\mathcal{A}_i\}$ are linear orderings from $\bigcup\{VD_\beta \mid \beta < \alpha\}$ and I is of the type $\omega, \omega^\star, \zeta$ or $\mathbf{n}$ for some $n < \omega$.*

*Define the class VD as the union of the VD_α's. The VD–***rank** *of a linear ordering $\mathcal{L} \in VD$, written $VD(\mathcal{L})$, is the least ordinal α such that $\mathcal{L} \in VD_\alpha$.*

Example 2.8 *Let $\mathcal{L}_1 = \Sigma\{\zeta + \mathbf{n} \mid n \in \omega\}$, $\mathcal{L}_2 = (\zeta \cdot \zeta) \cdot \zeta$. Then $VD(\mathcal{L}_1) = 2$, $VD(\mathcal{L}_2) = 3$ and $VD(\mathcal{L}_1 + \mathcal{L}_2) = 4$. In general, if $n = \max(VD(\mathcal{L}_1), VD(\mathcal{L}_2))$, then $n \leq VD(\mathcal{L}_1 + \mathcal{L}_2) \leq n + 1$.*

Example 2.9 *Let α, β be countable ordinals. Then $VD(\beta) \leq \alpha$ iff $\beta \leq \omega^\alpha$. In particular, $VD(\omega^\alpha) = \alpha$.*

Theorem 2.10 [14, Theorem 5.24] *A countable linear ordering $\mathcal{L}$ is scattered if and only if $\mathcal{L}$ is in VD.*

There is an alternative definition of ranking that we use in some of the proofs. We proceed with the definitions.

Definition 2.11 *A* **condensation map** *is a mapping c from L to non-empty intervals of L such that $c(y) = c(x)$ whenever $y \in c(x)$. The* **condensation** *of $\mathcal{L}$ is the linear order $c[\mathcal{L}]$ whose domain consists of the collection of non-empty intervals $c(x)$ for $x \in L$ ordered by $c(x) \ll c(y)$ if $\forall x_1 \in c(x) \forall y_1 \in c(y)(x_1 < y_1)$.*

Define the **iterated condensation** *c^α as a mapping from $\mathcal{L}$ to a set of non-empty intervals of $\mathcal{L}$ inductively as*

1. *$c^0(x) = \{x\}$ for all $x \in L$.*

2. $c^{\beta+1}(x) = \{y \in L \mid c(c^\beta(x)) = c(c^\beta(y))\}$,

3. $c^\lambda(x) = \bigcup\{c^\beta(x) \mid \beta < \lambda\}$ *for limit ordinal* λ.

Example 2.12 *As an illustration, we prove that every countable linear ordering can be represented as a dense sum of scattered linear orderings* (*Theorem 1.2*).

Proof The mapping $\{y \mid [x, y]$ is scattered$\}$, written $c_S(x)$, is a condensation since if $a \in c_S(x)$ then for all y, $[a, y]$ does not contain a dense subordering if and only if $[x, y]$ does not contain a dense subordering. Then $c_S[\mathcal{L}]$ is dense since for $c_S(x) \ll c_S(y)$, if there is no z with $c_S(x) \ll c_S(z) \ll c_S(y)$ then $[x, y]$ is scattered, contrary to assumption. Hence $c_S[\mathcal{L}]$ is a dense linear ordering and $\mathcal{L} = \sum\{a \mid a \in c[\mathcal{L}]\}$. Finally, note that each $a = c_S(x) \in c[\mathcal{L}]$ is scattered. □

An important condensation is $\{y \mid [x, y]$ is finite$\}$, written $c_{FC}(x)$ to which ones refers as a finite condensation. The idea here is that $c^1_{FC}(x)$ is the set of elements of $\mathcal{L}$ that are only finitely far away from x; $c^2_{FC}(x)$ is the set of elements of $\mathcal{L}$ that are in intervals of $c_{FC}[\mathcal{L}]$ which themselves are only finitely far away in $c_{FC}[\mathcal{L}]$ from the interval $c^1_{FC}(x)$. The least ordinal α such that $c^\beta_{FC}(x) = c^\alpha_{FC}(x)$ for all $x \in L$ and $\beta \geq \alpha$ is called the FC**–rank** of $\mathcal{L}$, written $FC(\mathcal{L})$. From now on, we write c for c_{FC}.

Example 2.13 *A linear order $\mathcal{L}$ is dense if and only if its FC–rank is 0. Moreover, $\mathcal{L}$ is scattered if and only if $c^\alpha[L] \simeq \mathbf{1}$ for some ordinal α.*

The following theorem connects FC-ranks and VD-ranks of scattered linear orderings.

Theorem 2.14 [14, Theorem 5.24] *If $\mathcal{L}$ is scattered then its VD–rank equals its FC–rank.*

If $A \subset L$ then we denote the condensation taking place within the set L by c and the condensation taking place relative to A by c_A. That is, c_A is just c with $\mathcal{A}$ replacing $\mathcal{L}$ in the definition. In this case we will also write $c_A(x)$, $c_A[\mathcal{A}]$ and $\ll_A$. Here are some useful properties, where the third (non-cited) one can be proven inductively.

Lemma 2.15 1. [14, Lemma 5.14] *If $\mathcal{L}$ is scattered and $M \subset L$ then $FC(\mathcal{M}) \leq FC(\mathcal{L})$.*

2. [14, Lemma 5.13 (2)] *$FC(c^\alpha(x)) \leq \alpha$ for every α, $x \in L$, and $c^\alpha(x)$ is a scattered interval of $\mathcal{L}$ for all α.*

3. *For every $x, y \in L$, if $[x, y]$ is scattered then $c^\alpha(x) = c^\alpha(y)$ if and only if $FC([x, y]) \leq \alpha$.*

4. [14, Exercise 5.12 (1)] *If I is an interval of $\mathcal{L}$ then for every α, $c^\alpha_I(x) = c^\alpha(x) \cap I$.*

3. Ranks of Automatic Linear Orderings

This section is devoted to the proof of Theorem 3.5. For this, we prove three propositions. Delhomme's idea [4] is to analyse the transition diagram of the automaton describing an ordinal, which we imitate in the proof of Theorem 3.4. As a matter of convenience, we introduce the following variation of rank.

Definition 3.1 *If $\mathcal{L}$ is scattered, define its VD_*–rank as being the least ordinal α such that $\mathcal{L}$ can be written as a* finite *sum of orderings of VD–rank $\leq \alpha$.*

For example, it is not hard to show that $VD(\omega) = VD_*(\omega) = 1$ and that $\omega 2 + 1$ has VD–rank 2 but VD_*–rank 1. We list two basic properties.

1. In general, $VD_*(\mathcal{L}) \leq VD(\mathcal{L}) \leq VD_*(\mathcal{L}) + 1$.

2. $c^\alpha[\mathcal{L}]$ is a finite linear order if and only if $VD_*(\mathcal{L}) \leq \alpha$.

Proposition 3.2 *Let $\mathcal{L} = (L, \leq)$ be a scattered linear ordering. Consider a finite partition of the domain $L = A_1 \cup A_2 \cup \cdots \cup A_k$. Then there exists some $1 \leq i \leq k$ with $VD_*(\mathcal{A}_i) = VD_*(\mathcal{L})$.*

Proof We prove the proposition for $k = 2$; the general case reduces to this case. Thus, assume that $A_0 \subset L$ and $A_1 = L \setminus A_0$. We need to show, by induction on $VD_*(\mathcal{L})$, that $VD_*(\mathcal{L}) = VD_*(\mathcal{A}_\epsilon)$ for some $\epsilon \in \{0, 1\}$. The case when $VD_*(\mathcal{L}) = 0$ or $VD_*(\mathcal{L}) = 1$ is checked easily. Assume that the proposition is true for all $\mathcal{L}$ such that $VD_*(\mathcal{L}) < \alpha$.

Suppose $VD_*(\mathcal{L}) = \alpha$. Then $\mathcal{L}$ is a finite sum of orders of VD–rank at most α. In particular, at least one of these must have VD–rank exactly α. Call it $\mathcal{M}$. Then $\mathcal{M}$ is an I–sum of linear orders $\{\mathcal{L}_i\}$ of VD–rank $< \alpha$, where I is of the type $\omega, \omega^\star, \zeta$ or $\mathbf{n}$ for some $n < \omega$. We may assume that $\mathcal{M}$ is chosen so that I is not finite, for if every such $\mathcal{M}$ were a finite sum of orders of VD–rank $< \alpha$, then $\mathcal{L}$ would have VD_*–rank $< \alpha$. So assume that I is infinite, say of type ω (the other infinite cases are similar).

For the first case, suppose that $\alpha = \beta + 1$. We can assume that there are infinitely many i such that $VD_*(\mathcal{L}_i) = \beta$, for otherwise we could write $\mathcal{M}$ as a finite sum of orders of VD–rank β, contrary to assumption. For each i let $A_{\epsilon,i} = L_i \cap A_\epsilon$, where $\epsilon \in \{0, 1\}$. Hence, applying the induction hypothesis to $\mathcal{L}_i$, we see that there is an $\epsilon \in \{0, 1\}$ and infinitely many j's such that $VD_*(\mathcal{A}_{\epsilon,j}) = VD_*(\mathcal{L}_j) = \beta$. Hence A_ϵ contains a subset which is an ω-sum of linear orders of VD_*-rank β. Therefore, $VD_*(\mathcal{A}_\epsilon) > \beta$, and so $VD_*(\mathcal{A}_\epsilon) = \alpha$ as required.

For the second case, suppose that α is a limit ordinal. So $\mathcal{M}$ is an ω–sum of linear orders $\{\mathcal{L}_i\}$ such the VD-rank of each $\mathcal{L}_i$ is less than α, and the supremum of the VD-ranks of $\mathcal{L}_i$ is α. Using the notation of the case above, and applying induction, we see that there is an $\epsilon \in \{0,1\}$ and infinitely may j's such that $VD_*(\mathcal{A}_{\epsilon,j}) = VD_*(\mathcal{L}_j)$, and the supremum of the VD_*-ranks of these $\mathcal{A}_{\epsilon,j}$'s is α. Then $VD_*(\mathcal{A}_\epsilon) = \alpha$ as required. □

Proposition 3.3 *Let $\mathcal{L}$ have FC–rank α. Then for every $\beta < \alpha$ there exists a closed scattered interval of $\mathcal{L}$ of rank $\beta + 1$.*

Proof Fix $\beta < \alpha$. Since $\mathcal{L}$ has FC–rank $> \beta$, by definition there is some $x \in L$ such that $c^\beta(x) \neq c^{\beta+1}(x)$. Pick $y \in c^{\beta+1}(x) \setminus c^\beta(x)$. Then $c^\beta(x) \neq c^\beta(y)$ and $c^{\beta+1}(x) = c^{\beta+1}(y)$. Recall that $c^\beta_{[x,y]}$ is the condensation mapping c^β within the interval $[x,y]$. Hence $c^\beta_{[x,y]}(x) \neq c^\beta_{[x,y]}(y)$ and $c^{\beta+1}_{[x,y]}(x) = c^{\beta+1}_{[x,y]}(y)$. The first fact implies that $VD([x,y]) > \beta$ and the second fact implies that $VD([x,y]) \leq \beta + 1$. Hence $VD([x,y]) = \beta + 1$ as required. □

Now we prove the following theorem:

Theorem 3.4 *The VD-rank of every automatic scattered linear ordering is finite.*

Proof Suppose $\mathcal{L}$ is automatic scattered linear over $\Sigma^\star$. Let $(Q_\leq, \iota_\leq, \Delta_\leq, F_\leq)$ be the 2–tape automaton recognising the ordering of $\mathcal{L}$. Let $(Q_A, \iota_A, \Delta_A, F_A)$ be the 3–tape automaton recognising the definable relation $\{(x,z,y) \mid x \leq z \leq y\}$. We assume the state sets Q_A and $Q_\leq$ are disjoint.

For $x, y \in L$ and $v \in \Sigma^\star$, define $[x,y]_v$ as the set of all $z \in L$ such that $x \leq z \leq y$ and z has prefix v. For $|v| \geq |x|, |y|$ define $I(x,v,y) \subset 2^{Q_A}$ and $J(x) \subset 2^{Q_\leq}$ as follows. $I(x,v,y)$ is the set of all states in Q_A reachable from the initial state ι_A after reading the convolution of (x,v,y), say $(x\diamond^n, v, y\diamond^m)$ where $n, m \geq 0$ are chosen so that the length of each component is exactly $|v|$. That is define $I(x,v,y) := \Delta_A(\iota_A, (x,v,y)^\diamond)$. Similarly, define $J(v) := \Delta_\leq(\iota_\leq, (v,v)^\diamond)$. Write $K(x,v,y)$ for the ordered pair $(I(x,v,y), J(v))$.

Now if $K(x,v,y) = K(x',v',y')$, then $[x,y]_v$ is isomorphic to $[x',y']_{v'}$ via the map $vw \mapsto v'w$ for $w \in \Sigma^\star$. Indeed, the domains are isomorphic since $vw \in [x,y]_v$ if and only if

$$\Delta_A(\Delta_A(\iota_A, (x,v,y)^\diamond), w) \cap F_A \neq \emptyset$$

if and only if

$$\Delta_A(\Delta_A(\iota_A, (x',v',y')^\diamond), w) \cap F_A \neq \emptyset$$

if and only if $v'w \in [x',y']_{v'}$.

The map preserves the ordering since for $w_1, w_2 \in \Sigma^\star$ such that $vw_1, vw_2 \in [x,y]_v$ and $v'w_1, v'w_2 \in [x',y']_{v'}$ we have $vw_1 \leq vw_2$ if and only if

$$\Delta_\leq(\Delta_\leq(\iota_\leq, (v,v)^\diamond), (w_1,w_2)^\diamond) \cap F_\leq \neq \emptyset$$

if and only if

$$\Delta_\leq(\Delta_\leq(\iota_\leq, (v',v')^\diamond), (w_1,w_2)^\diamond) \cap F_\leq \neq \emptyset$$

if and only if $v'w_1 \leq v'w_2$.

Hence then number of isomorphism types of the form $[x,y]_v$ for $|v| \geq |x|, |y|$ is bounded by the number of distinct sets $K(x,v,y)$ which is at most $2^{Q_A + Q_\leq}$, denoted by d. In particular there are at most d many VD_*–ranks among such intervals of the form $[x,y]_v$. Now let $[x,y]$ be a closed interval of $\mathcal{L}$. Set $n = \max\{|x|, |y|\}$ and partition $[x,y]$ into the set $[x,y] \cap \Sigma^{<n}$ and the finitely many sets of the form $[x,y]_v$ where $|v| = n$. By Proposition 3.2, one of these intervals has the same VD_*-rank as the interval $[x,y]$. Suppose the VD–rank of $[x,y]$ is greater than d. By Proposition 3.3 there would be more than d intervals of different VD– and hence VD_*–ranks. So among intervals of the form $[x,y]_v$, where $|v| \geq |x|, |y|$, there would be more than d many VD_*–ranks, a contradiction. We conclude that the VD–rank of $[x,y]$ is at most d. So for every $x, y \in L$, $c^d(x) = c^d(y)$ so $VD(\mathcal{L}) \leq d$ as required. □

As a corollary of the theorem just proved we derive the following result for all automatic linear orderings:

Theorem 3.5 *If $\mathcal{L}$ is an automatic linear order, then its FC–rank is finite.*

Proof Let $(Q_\leq, \iota_\leq, \Delta_\leq, F_\leq)$ be the 2–tape automaton recognising the ordering of $\mathcal{L}$. Let $(Q_A, \iota_A, \Delta_A, F_A)$ be the 3–tape automaton recognising the definable relation $\{(x,z,y) \mid x \leq z \leq y\}$. Now consider an interval $[a,b]$ contained in the maximal scattered linear order containing a given element x of $\mathcal{L}$. Consider the constant d defined in the proof of the previous theorem. Note that this constant does not depend on the interval $[a,b]$ chosen. Therefore the FC-rank of the interval is at most d. We conclude that the FC–rank of the maximal scattered order is at most d, and so the FC–rank of $\mathcal{L}$ is at most d. □

The results above can now be applied to show that the isomorphism problem for automatic ordinals is decidable. Contrast this with the fact that the isomorphism problem for computable ordinals is Π^1_1-complete. Recall that by Cantor's normal form theorem if α is an ordinal then it can be uniquely decomposed as $\omega^{\alpha_1} n_1 + \omega^{\alpha_2} n_2 + \ldots + \omega^{\alpha_k} n_k$, where $\alpha_1, \alpha_2, \ldots, \alpha_k$ are ordinals satisfying $\alpha_1 > \alpha_2 > \ldots > \alpha_k$ and $k, n_1, n_2, \cdots, n_k$ are natural numbers. Our proof of deciding the isomorphism problem for automatic ordinals based on the fact that the Cantor normal form can be extracted from automatic presentations of ordinals.

Theorem 3.6 *If α is an automatic ordinal then its normal form is computable from an automatic presentation of α.*

Proof The automatic presentation for α is given by a regular set $R \subseteq \Sigma^*$ for some alphabet Σ and an automaton for the ordering $\leq_{ord}$ on R. Recall that the unknown ordinal represented by $(R, \leq_{ord})$ is of the form $\alpha = \omega^m n_m + \omega^{m-1} n_{m-1} + \ldots + \omega^2 n_2 + \omega n_1 + n_0$ where $m, n_m, n_{m-1}, \ldots, n_1, n_0$ are natural numbers. Now one can compute the values $m, n_0, n_1, \ldots$ by the following algorithm.

1. **Input** the presentation $(R, \leq_{ord})$.
2. Let $D = R, m = 0, n_m = 0$.
3. **While** $D \neq \emptyset$ **Do**
4. **If** D has a maximum u

   Then Let $n_m = n_m + 1$, let $D = D - \{u\}$.

   Else Let $L \subseteq D$ be the subset of limit ordinals in D; that is L is the set of all $x \in D$ with no predecessor in D. Replace D by L, let $m = m + 1$, let $n_m = 0$.
5. **End While**
6. **Output** the formula

   $$\omega^m n_m + \omega^{m-1} n_{m-1} + \ldots + \omega^2 n_2 + \omega n_1 + n_0$$

   using the current values of $m, n_0, \ldots, n_m$.

Since the first order theory of an automatic structure is decidable, each step in the algorithm is computable. Removing the maximal element from D reduces the ordinal represented of D by 1 while the corresponding n_m is increased by 1. Replacing D by the set of its limit ordinals is like dividing the ordinal represented by D by ω, so that the next coefficient can start to be computed. Based on this it is easy to verify that the algorithm computes the coefficients $n_0, n_1, \ldots$ in this order. The algorithm eventually terminates since m is bounded by the finite bound on the VD-rank of the ordinal. □

The following corollary is immediate.

Corollary 3.7 *The isomorphism problem for automatic ordinals is decidable.*

Compare this with the fact that the isomorphism problem for automatic structures, and even permutation structures, is not decidable [3],[8]. We do not know if the isomorphism problem for automatic linear orders is decidable.

We would like to say a few words on the problem of characterising automatic linear orders. We already have a characterisation of the isomorphism types of linear orders presentable over a unary domain.

Theorem 3.8 [1, 12] *A linear ordering is automatically presentable over a unary alphabet if and only if it is a finite sum of linear orders of VD–rank ≤ 1.*

The non-unary case is far more involved. A full characterisation of the automatic scattered linear orders would further refine Theorem 3.5. For example $\mathcal{L} = \Sigma_{i \in \mathbb{N}}(\mathbb{Z} + \mathbf{n}_i)$, where $f(i) = n_i$ is function from ω into ω, has FC–rank 2. Note that if f is a non-recursive function then $\mathcal{L}$ is not automatic; otherwise the decidability of $\mathcal{L}$ could be used to compute f. The following examples indicate the complexities involved in the general case.

Example 3.9 [8] *Let $\mathcal{E}$ is an automatic equivalence structure, with finite equivalence classes $\{B_i\}$ for $i \in \omega$. Then the linear order $\mathcal{L}_{\mathcal{E}} = \Sigma\{\eta + \mathbf{n}_i \mid i \in \omega\}$ of F-rank 2, where n_i is the cardinality of B_i, is automatically presentable.*

Hence a characterisation of automatic equivalence structures would follow from one of automatic linear orders. But automatic equivalence structures already exhibit the following behaviour.

Example 3.10 [8] *Consider equivalence structures $\mathcal{E}_1 = (\{0,1\}^*, E)$ and $\mathcal{E}_2 = (0^\star 1^\star, E)$, where $E(x,y)$ if $|x| = |y|$ and x, y belong to the same domain. In E_1 for each n there is a unique equivalence class of size 2^n. In $\mathcal{E}_2$ for each n there is a unique equivalence class of size $(n+1)(n+2)/2$. In general, for every function f that is either a polynomial whose coefficients are positive integers, or an exponential function k^{an+b}, where $k \geq 2$ and a, b are fixed positive integers, there exists an automatic equivalence relation $\mathcal{E}$ with a unique finite equivalence class of size $f(n)$ for every $n \geq 1$.*

4. Automatic trees

A *tree* $\mathcal{T} = (T, \preceq)$ is a partial order that has a least element r, called the root, and in which $\{y \in T \mid y \preceq x\}$ is a finite linear order for each $x \in T$. Write $x \| y$ if $x \not\preceq y$ and $y \not\preceq x$. The set $S(x)$ of immediate successors of x is defined as $S(x) = \{y \in \mathcal{T} \mid x \prec y \wedge \forall z(x \preceq z \preceq y \Rightarrow z \in \{x, y\})\}$. A tree $\mathcal{T}$ is **finitely branching** if $S(x)$ is finite for each $x \in T$. A **path** of a tree $(T, \preceq)$ is a subset $P \subseteq T$ which is linearly ordered, closed downward (that is, whenever $y \in P$ and $x \preceq y$ then $x \in P$), and maximal (under set-theoretic inclusion) with these properties.

Let Σ be the underlying alphabet and let $\leq_{ll}$ be the length-lexicographic order of Σ^* where $x \leq_{ll} y$ if either $|x| < |y|$ or $|x| = |y|$ but x lexicographic before y. For example, $\lambda <_{ll} 0 <_{ll} 1 <_{ll} 00 <_{ll} 01 <_{ll} \ldots$ in the case that $\Sigma = \{0, 1\}$. Thus, if tree $\mathcal{T}$ is automatic over the alphabet Σ, and hence $T \subseteq \Sigma^*$, then the length-lexicographic order on Σ^* is inherited by each set $S(x)$. This permits us to talk about the first, second, third, ... successor of x.

Trees have been studied intensively. If one considers Turing machines instead of finite automata, there are trees which have infinite paths, but no hyperarithmetic one, in particular no recursive infinite path. Furthermore, even finitely branching trees might have infinite paths but none of them is recursive. In contrast to this, the next result states that every automatic tree, not necessarily finitely branching, either has a regular infinite path or does not have an infinite path at all.

Theorem 4.1 *If an automatic tree has an infinite path, then it has a regular infinite path.*

Proof Let $\mathcal{T} = (T, \preceq)$ be an automatic tree with some infinite path where T is a regular subset of Σ^* for a finite alphabet Σ. The proof consists of the following two parts.

1. Given an automatic tree $\mathcal{T} = (T, \preceq)$, the set T' of all nodes in T which are on an infinite path is regular.

2. There is a definable infinite path P on T' in the language expanded by the automatic length-lexicographic order. Hence P is regular.

1. To prove that T' is regular it suffices to show that:

- There is an alphabet Δ, a function $\pi : \Delta \to \Sigma_\diamond$ and a Büchi automaton $\mathcal{B}$ such that $a_0 \dots a_n \in T'$ if and only if there is an infinite sequence $c_0 c_1 \dots \in \Delta^\infty$ accepted by $\mathcal{B}$ and $a_0 = \pi(c_0), a_1 = \pi(c_1), \dots, a_n = \pi(c_n), \diamond = \pi(c_{n+1})$.

Choose the alphabet Δ as $\Sigma_\diamond \times \Sigma$ where π is the projection from Δ onto its first coordinate, that is, $\pi(a, b) = a$. Say that a word x *is on* $c_0 c_1 \dots$, where each c_i is $(a_i, b_i) \in \Sigma_\diamond \times \Sigma$, iff there exist $m, n \in \mathbb{N}$ such that

either $m = 0$, $x = a_0 a_1 \dots a_n$ and $a_{n+1} = \diamond$

or $n \geq m > 0$, $x = b_0 b_1 \dots b_{m-1} a_m a_{m+1} \dots a_n$, $a_{m-1} = \diamond$ and $a_{n+1} = \diamond$.

In the first case we say that x is the *first word on* $c_0 c_1 \dots$. Consider the set of all sequences $(a_0, b_0)(a_1, b_1) \dots \in \Delta$ such that there are infinitely many words on the sequence and the words on the sequence generate an infinite path of T. More formally,

- $\exists^\infty n (a_n = \diamond)$;

- if y, z are on $(a_0, b_0)\,(a_1, b_1)\,\dots$ and $|y| \leq |z|$ then $y \preceq z$ and $y, z \in T$.

There is a Büchi automaton $\mathcal{B}$ accepting such sequences because the orderings $\preceq$ and length-comparison are automatic and T is regular.

We prove that $x \in T'$ if and only if x is the first word on some sequence $c_0 c_1 \dots$ satisfying the two conditions. The reverse implication is clear. For the forward implication let $x \in T$ be given and P be an infinite path witnessing that $x \in T'$. Define the sequences $a_0 a_1 \dots$ and $b_0 b_1 \dots$ described below.

1. **Choose** $n, a_0, a_1, \dots, a_n$ such that $x = a_0 a_1 \dots a_n$. **Let** $a_{n+1} = \diamond$.

2. **Let** $m = 0$. **Let** $y = x$.

3. **Find** $b_m b_{m+1} \dots b_{n+1}$ such that infinitely many nodes in P extend $b_0 b_1 \dots b_{n+1}$ as strings.

4. **Update** $m = n + 2$.

5. **Find** a new value for n and $a_m a_{m+1} \dots a_n$ such that $n \geq m$, the node $z = b_0 b_1 \dots b_{m-1} a_m a_{m+1} \dots a_n$ is in P and $y \preceq z$. **Let** $a_{n+1} = \diamond$.

6. **Let** $y = z$. **Go to** 3.

Note that it is an invariant of the construction that whenever the algorithm comes to Step 3, either $m = 0$ or infinitely many nodes in P extend the string $b_0 b_1 \dots b_{m-1}$. As there are only finitely many choices for the new part $b_m b_{m+1} \dots b_{n+1}$, one can choose this part such that still infinitely many nodes in P extend $b_0 b_1 \dots b_{n+1}$ as a string. In Step 4, m is chosen such that the precondition of Step 3 holds again and b_m is the first of the b-symbols not yet defined. For every $y \in P$ it holds that almost all nodes z in P satisfy $y \preceq z$. Furthermore, for every finite length l, almost all nodes in P are represented by strings longer than l. Thus one can find a node z as specified in Step 5 and the algorithm runs forever defining the infinite sequence $(a_0, b_0)(a_1, b_1) \dots$ in the limit. In particular, such a sequence exists. It is not required that the sequence can be constructed effectively since the path P might not even be recursive.

2. Now we give a first order definition of the lexicographically least infinite path. Recall that the length-lexicographic order $\leq_{ll}$ on Σ^* is automatic and therefore one can add $\leq_{ll}$ to the structure $\mathcal{T}'$ without loosing the property that it is an automatic structure. Now define the leftmost infinite path P with respect to the length-lexicographic order of the successors of any node. P contains those nodes x for which every $y \prec x$ satisfies that $\forall z, z' \in S(y)\,[z \preceq x \Rightarrow z \leq_{ll} z']$. This means, that the unique node $z \in S(y)$ which is below x is just the length-lexicographically least element of $S(y)$. Since the length-lexicographic ordering of Σ^* is a well-ordering (of type ω), this minimum always exists. □

Remark 4.2 *In the case that the automatic tree $\mathcal{T} = (T, \preceq)$ is infinite and finitely branching, one can simplify the first part of the proof. The reason is that T' is definable by formula $x \in T' \Leftrightarrow (\exists^\infty y)\, [x \preceq y]$. If there are infinitely many nodes above x, x is by König's Lemma on an infinite path. If there are only finitely many nodes above x, these nodes trivially cannot contain an infinite path. The second part is proven in the same way as above.*

From Theorem 4.1, we see that if an automatic tree has *finitely* many infinite paths, then each is regular. We generalise this to trees with *countably* many infinite paths (Theorem 4.4). The proof relies on associating the Kleene-Brouwer-ordering with trees.

Definition 4.3 [13] *Let $(T, \preceq)$ be a tree and $\leq_{ll}$ be the lexicographic order induced by the presentation of T as a subset of Σ^*. Let x, y be nodes on T. Then $x \leq_{kb} y$ iff either $y \preceq x$ or there are u, v, w such that $v, w \in S(u)$, $v \preceq x$, $w \preceq y$ and $v \leq_{ll} w$.*

Theorem 4.4 *If an automatic tree has countably many infinite paths then every infinite path in it is regular.*

Proof Let $(T_1, \preceq)$ be an automatic tree with countably many infinite paths. Let Σ be the alphabet and $\leq_{ll}$ be the length-lexicographic ordering. Furthermore $\leq_{kb}$ is the Kleene-Brouwer ordering derived from $\leq_{ll}$ and T_1 as defined in Definition 4.3. Inductively, for $n = 1, 2, \ldots$, let

$$\begin{aligned} T'_n &= \{x \in T_n \mid \exists \text{ inf. path } P \text{ of } T_1 \\ &\qquad (x \in P \wedge P \cap T_n \text{ is infinite})\}; \\ T_{n+1} &= \{x \in T'_n \mid \forall y \in T'_n \exists z \in T'_n \\ &\qquad (y \leq_{kb} x \vee x <_{kb} z <_{kb} y)\}. \end{aligned}$$

The set T'_n contains those nodes which are on an infinite path of T_1 which has an infinite intersection with T_n. The set T_{n+1} contains those nodes of T'_n which are the infimum of the properly above nodes in T'_n. The structures $(T_n, \preceq, \leq_{kb})$ and $(T'_n, \preceq, \leq_{kb})$ are automatic iff the corresponding sets T_n and T'_n are regular subsets of Σ^*. The set T_1 is regular. If T_n is regular, so is T'_n by the construction in the first part of the proof of Theorem 4.1. T_{n+1} is obtained from T'_n using a first-order formula and therefore also regular. So one has by induction that $(T_1, T'_1, T_2, T'_2, \ldots, \preceq, \leq_{kb})$ is an automatic structure.

Now it is first proven that there is an n for which T_n is finite. Let $c^1_F(A, x) = c^1_F(A, y)$ denote that there are only finitely many elements of A between x and y with respect to $\leq_{kb}$, for $m \geq 1$ the notion $c^m_F(A, x) = c^m_F(A, y)$ is the iterated versions of it. If $x, y \in T'_n$ and $c_F(T'_n, x) = c_F(T'_n, y)$, then there are only finitely many elements between them and one of the numbers x, y cannot be the limit inferior of an infinite descending chain in $(T'_n, \leq_{kb})$. Thus at most one of the elements x, y is in T_{n+1}. Iterating the argument, it follows that for every $x, y \in T_1$ with $c^n_F(T_1, x) = c^n_F(T_1, y)$, at most one of the elements x, y is in T_{n+1}. Since the structure $(T_1, \leq_{kb})$ is an automatic linear ordering, its FC–rank is finite and thus some T_n contains at most one element. In particular $T_n \cap P$ is finite for all infinite paths P of T_1.

So there is, for every infinite path P of T_1, a maximal number n such that $T_n \cap P$ is infinite. It follows from the definition of T'_n that $T'_n \cap P = T_n \cap P$ and so, $T'_n \cap P$ is also infinite. Now take a node $x \in T'_n \cap P$ such that no node $z \succeq x$ is in $T_{n+1} \cap P$. Let y be the least element of $T'_n \cap P$ with respect to $\preceq$ satisfying $x \prec y$.

Assume that there would be an another $z \succ x$ in T'_n which satisfies $y <_{kb} z <_{kb} x$. Then $y \| z$. But since $z \in T'_n$ there is an infinite path P' of T_1 such that $P' \cap T'_n$ is infinite and contains z. So there is a node $z' \succ z$ in $P' \cap T'_n$. This node satisfies $y <_{kb} z' <_{kb} z$. Therefore y is in T'_n the infimum of the nodes $\{z \in T'_n \mid y <_{kb} z\}$ and y would be in T_{n+1} in contradiction to the choice of n, x, y.

Thus it holds for all $y \in P \cap T'_n$ and $z \in T'_n$, whenever $y \leq_{kb} z \leq_{kb} x$ then $z \in P$. Furthermore, every $y \in P \cap T'_n$ with $x \preceq y$ satisfies that the set $\{z \mid y \leq_{kb} z \leq_{kb} x\}$ is finite. So the first-order definable set $P_{x,n} = \{y \succeq x \mid y \in T'_n \wedge \neg \exists^\infty z (y \leq_{kb} z \leq_{kb} x)\}$ is equal to $\{y \succeq x \mid y \in P \cap T'_n\}$. It follows that P is the downward closure with respect to $\preceq$ in T_1 of $P_{x,n}$ and P is regular. □

In Section 5 it is shown that every automatic tree has finite Cantor-Bendixson rank. This would permit to simplify the constructions of the sets $T'_1, T_2, T'_2, \ldots$ which then are subtrees still satisfying that almost all of these sets are empty.

$$\begin{aligned} T'_n &= \{x \in T_n \mid \exists \text{ inf. path} P \text{ of } T_n (x \in T_n)\}; \\ T_{n+1} &= \{x \in T'_n \mid \exists y, z \in T'_n (x \preceq y \wedge x \preceq z \wedge y \| z)\}. \end{aligned}$$

If T_1 is furthermore finitely branching, the first condition is equivalent to

$$T'_n = \{x \in T_n \mid \exists^\infty y \in T_n (x \preceq y)\}$$

so that all trees T_n, T'_n are first-order definable in the language $(T_1, \preceq)$. For every infinite path P of T_1 there is a maximal n such that P is on T_n. If a is a node on P which is sufficiently large than all $b \in T_n$ satisfy that $b \in P$ iff $a \preceq b \vee b \preceq a$. Only finitely many of the trees, say T_1, T_2, T_3, T_4, have infinite paths and the formula

$$\begin{aligned} \phi(a, b) &= (a \in T'_1 \wedge a \notin T_2 \wedge b \in T'_1 \wedge (b \preceq a \vee a \preceq b)) \\ &\vee\ (a \in T'_2 \wedge a \notin T_3 \wedge b \in T'_2 \wedge (b \preceq a \vee a \preceq b)) \\ &\vee\ (a \in T'_3 \wedge a \notin T_4 \wedge b \in T'_3 \wedge (b \preceq a \vee a \preceq b)) \\ &\vee\ (a \in T'_4 \wedge a \notin T_5 \wedge b \in T'_4 \wedge (b \preceq a \vee a \preceq b)) \end{aligned}$$

tells for every infinite path P and almost every $a \in P$ which nodes b are on P. So $\{b \mid \phi(a, b)\}$ is either an infinite path of T_1 or empty.

In the case that $(T_1, \preceq)$ is infinite branching, one modifies the tree. Recall that the set $S(x)$ of immediate successors of $x \in T$:

$$y \in S(x) \Leftrightarrow x \prec y \wedge \forall z (z \not\prec y \vee x \not\prec z).$$

Furthermore, let $\leq_{ll}$ be the length-lexicographic ordering on the underlying set Σ^*. Now let r be the root of the tree $(T_1, \preceq)$ and define the new ordering $\preceq'$ by letting $x \preceq' y$ if and only if

$$x = r \vee \exists v, w \in T_1\, (x, w \in S(v) \wedge x \leq_{ll} w \wedge w \preceq y)$$

The tree $(T_1, \preceq')$ is finitely branching. Every infinite branch P' of $(T_1, \preceq')$ are generated either by an infinite branch P of $(T_1, \preceq)$ or by an infinite set of the form $S(x)$ where the set $S(x)$ refers to the immediate successors of an $x \in T_1$ with respect to $\preceq$. Let ϕ' be the formula for the infinite branches of $(T_1, \preceq')$. If P' is an infinite branch of T_1 which is generated by P, then an $x \in P'$ is also in P iff there is an $y \in P'$ such that $x \prec y$. Furthermore if P' is generated by a set $S(u)$ then almost all $x \in P'$ do not have an $y \in P'$ with $x \prec y$ since these x are in $S(u)$. Thus the following formula ϕ derived from ϕ' has the desired properties:

$$\phi(a,b) = \phi'(a,b) \wedge \exists^\infty x \exists y (\phi(a,x) \wedge \phi(a,y) \wedge b \preceq x \preceq y).$$

Note furthermore, that the quantification "$\exists^\infty$" enforces as a side-effect that $\phi(a,b)$ defines only infinite sets and that thus one has the following result.

Theorem 4.5 *If $(T, \preceq)$ is an automatic tree with countably many infinite paths, then there is a formula ϕ such that the sets $P_a = \{b : \phi(a,b)\}$ satisfies the following conditions:*

- *If P_a is not empty, then P_a is an infinite path of $(T, \preceq)$ containing a;*
- *Every infinite path of $(T, \preceq)$ is equal to some set P_a.*

The option that P_a can be empty in the definition above cannot be avoided since a might be in T but not on an infinite path of T.

5. Cantor-Bendixson Rank of Automatic Trees

In this section it is shown that every automatic tree with countably many infinite paths has finite Cantor-Bendixson Rank. For this, we need some additional notation.

We write $[\mathcal{T}]$ for the set of infinite paths of $\mathcal{T}$ and if $x \in T$ and $p = (p_i) \in [\mathcal{T}]$ write $x \prec p$ if $x = p_i$ for some i. Define the extendible part $E(\mathcal{T})$ of $\mathcal{T}$ as $\{x \in T \mid \exists p \in [\mathcal{T}], x \prec p\}$. Note that $E(\mathcal{T}) = \{x \in T \mid \exists^\infty y, x \prec y\}$ if $\mathcal{T}$ is finitely branching. Let $F(\mathcal{T})$ be the domain $\{x \in E(\mathcal{T}) \mid \exists y, z \in E(\mathcal{T}), y, z \succ x \text{ and } y \| z\}$. Write $d(\mathcal{T})$ for the tree resulting by restricting $\mathcal{T}$ to $F(\mathcal{T})$. In words, a node $x \in \mathcal{T}$ is in $d(\mathcal{T})$ if and only if there are at least two distinct infinite paths in $\mathcal{T}$ above x. For each ordinal α define the iterated operation $d^\alpha(\mathcal{T})$ inductively as follows.

1. $d^0(\mathcal{T}) = \mathcal{T}$.
2. $d^{\alpha+1}(\mathcal{T})$ is $d(d^\alpha(\mathcal{T}))$.
3. If α is a limit ordinal, then $d^\alpha(\mathcal{T})$ is $\cap_{\beta<\alpha} d^\beta(\mathcal{T})$.

Definition 5.1 [9] *The* **Cantor-Bendixson Rank** *of a tree $\mathcal{T}$ (written $CB(\mathcal{T})$) is the least ordinal α such that $d^\alpha(\mathcal{T}) = d^{\alpha+1}(\mathcal{T})$.*

Lemma 5.2 *1. $CB(\mathcal{T})$ is a countable ordinal.*

2. Suppose $\mathcal{T}$ has countably many infinite paths.

(a) $\alpha = CB(\mathcal{T})$ is 0 or a successor ordinal and $d^\alpha(\mathcal{T})$ is the empty tree.

(b) $CB(\mathcal{T}) = 0$ if and only if $\mathcal{T}$ is the empty tree.

(c) $CB(\mathcal{T}) = 1$ if and only if $\mathcal{T}$ is non-empty and contains at most one infinite path.

Proof For each β let $x_\beta \in d^\beta(\mathcal{T}) \setminus d^{\beta+1}(\mathcal{T})$. Since T is countable, and $\alpha \neq \beta$ implies that $x_\alpha \neq x_\beta$, the set of ordinals β such that $d^\beta(\mathcal{T}) \setminus d^{\beta+1}(\mathcal{T}) \neq \emptyset$ is also countable. Hence its least upper bound, a countable ordinal, say α, is $CB(\mathcal{T})$. If $d^\alpha(\mathcal{T})$ is not the empty tree, then every element of $d^\alpha(\mathcal{T})$ has at least two distinct infinite paths above it. In particular, $d^\alpha(\mathcal{T})$ embeds the full binary tree, so $\mathcal{T}$ has uncountably many infinite paths. Finally suppose $\alpha > 0$ is a limit ordinal. Then $d^\alpha(\mathcal{T})$ is non-empty since the root of $\mathcal{T}$ is in $d^\beta(\mathcal{T})$ for every $\beta < \alpha$. Hence the CB–rank of $\mathcal{T}$ is 0 or a successor ordinal. The last two items follow directly from the definition of d. $\square$

The next definition associates a linear ordering $\mathcal{L}_T$ with a tree $\mathcal{T}$.

Definition 5.3 *Let $(T, \preceq)$ a tree and recall that $\leq_{ll}$ is a linear order of type ω on T. Let $x, y \in T$. Then $x <_{lt} y$ if either $x \prec y$ or there are u, v, w such that $v, w \in S(u)$, $v \preceq x$, $w \preceq y$ and $v <_{ll} w$. Write $\mathcal{L}_T$ for the structure $(T, <_{lt})$.*

Then $\mathcal{L}_T$ is a linear ordering and is definable in $(\mathcal{T}, \leq_{ll})$.

Theorem 5.4 *The CB-rank of an automatic finitely branching tree with countably many infinite paths is finite.*

Proof Suppose $\mathcal{T}$ is finitely branching with countably many infinite paths. We now prove that $\mathcal{L}_T$ is scattered and is a finite sum of orders of VD–rank $\leq CB(\mathcal{T})$. In this case, if $\mathcal{T}$ is automatic then so is $\mathcal{L}_T$, which by theorem 3.4 has finite VD–rank. Then $CB(\mathcal{T})$ must also be finite as required.

We deal with the base cases first. If $CB(\mathcal{T}) = 0$ then $\mathcal{T}$ is the empty tree and so $\mathcal{L}_T$ is the empty linear order, which satisfies the conclusion. If $CB(\mathcal{T}) = 1$ then $\mathcal{T}$ is non-empty and contains at most one infinite path. By the definition of $<_{lt}$, L_T has order type $\mathbf{n}, \omega, \omega + \mathbf{n}$ or $\omega + \omega^\star$. In every case it is scattered and a finite sum of orders of VD–rank ≤ 1.

Now let $\alpha = CB(\mathcal{T}) > 1$. By the previous lemma, $\alpha = \beta + 1$ for some $\beta > 0$. For $x \in T$ let $T(x)$ be the subtree of $\mathcal{T}$ rooted at x, that is $\{y \in T \mid x \preceq y\}$. Define $X = \{x \in T \mid CB(T(x)) = \alpha\}$. Then $X \neq \emptyset$ since the root of $\mathcal{T}$ is in X. Further $X = d^\beta(\mathcal{T})$. Indeed if $x \in X$ then in particular $x \in d^\beta(T(x))$ and so $x \in d^\beta(\mathcal{T})$. Conversely if $x \notin X$ then $d^\gamma(T(x)) = \emptyset$ for some $\gamma \leq \beta$. Hence $x \notin d^\gamma(\mathcal{T})$ and in particular $x \notin d^\beta(\mathcal{T})$.

Hence $CB(X) = 1$ and so X contains at most one infinite path. For $x \in X$, consider an immediate successor of x in $\mathcal{T}$, that is not itself in X, say $y \in S(x) \setminus X$. Then by definition of X, $CB(T(y)) \leq \beta$. By the induction hypothesis $L_{T(y)}$ is scattered and is a finite sum of orders of VD rank $\leq \beta$. If X is finite then L_T is a finite sum of the $L_{T(y)}$'s, of which there are finitely many since T is finitely branching. Hence L_T is scattered and satisfies the conclusion. If X is infinite, let (x_i) be the unique infinite path in X. Then from the definition of $<_{lt}$,

$$L_T = \sum_{i \in \omega} (x_i + L_i) + \sum_{j \in \omega^\star} R_j,$$

where every L_i or R_j is some $T(y)$. Hence L_T is scattered and is a finite sum of orders of VD–rank $\leq \beta + 1$ as required. □

Theorem 5.5 *The CB-rank of an automatic tree with countably many infinite paths is finite.*

Proof As in the text before Theorem 4.5, one defines from the given tree $(T, \preceq)$ the tree $(T, \preceq')$ such that $x \preceq' y$ iff

$$x = r \vee \exists v, w \in T\, (x, w \in S(v) \wedge x \leq_{ll} w \wedge w \preceq y);$$

where r is the root of T, $\leq_{ll}$ the length-lexicographic order and $S(v)$ the set of immediate successors of v with respect to $\preceq$. The finitely branching tree $(T, \preceq')$ also has countably many infinite paths and therefore finite CB–rank. Let U and U' be the topological spaces of the sets of infinite paths of $(T, \preceq)$ and $(T, \preceq')$, respectively. Every infinite path P of $(T, \preceq)$ generates an infinite path P' of $(T, \preceq')$ and one can show that the topological spaces U and $\{P' \in U' \mid \exists P \in U$ $(P$ generates $P')\}$ are homeomorphic. Thus, the CB–rank of U' (as a topological space) is an upper bound of that of U and the same holds for the corresponding trees (as the CB–rank of a tree is the same as the CB–rank of the corresponding topological space). Thus, the Cantor-Bendixson rank of the tree $(T, \preceq)$ is also finite. □

Acknowledgement The authors thank Wolfgang Merkle for useful discussions and the anonymous referees for helpful comments regarding presentation.

References

[1] A. Blumensath. *Automatic Structures*. Diploma thesis, RWTH Aachen, 1999.

[2] A. Blumensath and E. Grädel. Automatic structures. In *15th Symposium on Logic in Computer Science (LICS' 00)*, pages 51–62, June 2000.

[3] A. Blumensath and E. Grädel. Finite presentations of infinite structures : Automata and interpretations. In *Proc. 2nd Int. Workshop on Complexity in Automated Deduction(CIAD)*, 2002.

[4] C. Delhomme. Non-automaticity of ω^ω, 2001. manuscript.

[5] C. Delhomme, V. Goranko, and T. Knapik. Automatic linear orderings, 2003. manuscript.

[6] S. Eilenberg, C. Elgot, and J. Shepherdson. Sets recognised by n–tape automata. *Journal of Algebra*, 13(4):447–464, 1969.

[7] D. B. H. 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, 1992.

[8] H. Ishihara, B. Khoussainov, and S. Rubin. Some results on automatic structures. In *17th Symposium on Logic in Computer Science (LICS' 02)*, pages 235–242, July 2002.

[9] A. Kechris. *Classical Descriptive Set Theory*. Springer-Verlag, 1995.

[10] B. Khoussainov and A. Nerode. Automatic presentations of structures. *Lecture Notes in Computer Science*, 960:367–392, 1995.

[11] B. Khoussainov and S. Rubin. Automatic structures: Overview and future directions. *Journal of Automata, Languages and Combinatorics. Selected papers of Weighted Automata : Theory and Applications (2002)*.

[12] B. Khoussainov and S. Rubin. Graphs with automatic presentations over a unary alphabet. *Journal of Automata, Languages and Combinatorics*, 6(4), 2001.

[13] H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, 1967.

[14] J. Rosenstein. *Linear Orderings*. Academic Press, 1982.