Automatic Structures: Richness and Limitations

Bakhadyr Khoussainov*, André Nies†, Sasha Rubin‡ and Frank Stephan§

Abstract

We study the existence of automatic presentations for various algebraic structures. An automatic presentation of a structure is a description of the universe of the structure by a regular set of words, and the interpretation of the relations by synchronised automata. Our first topic concerns characterising classes of automatic structures. We supply a characterisation of the automatic Boolean algebras, and it is proven that the free Abelian group of infinite rank, as well as certain Fraïssé limits, do not have automatic presentations. In particular, the countably infinite random graph and the random partial order do not have automatic presentations. Furthermore, no infinite integral domain is automatic. Our second topic is the isomorphism problem. We prove that the complexity of the isomorphism problem for the class of all automatic structures is Σ^1_1-complete.

1 Introduction

Classes of infinite structures with nice algorithmic properties (such as decidable model checking) are of increasing interest in a variety of fields of computer science. For instance the theory of infinite state transition systems concerns questions of symbolic representations, model checking, specification and verification. Also, string query languages in databases may be captured by (decidable) infinite string structures. In these and other areas there has been an effort to

*B. Khoussainov, Department of Computer Science, University of Auckland, New Zealand. Email: bmk@cs.auckland.ac.nz. The first and the second authors' research was partially supported by the Marsden Fund of New Zealand.

†A. Nies, Department of Computer Science, University of Auckland, New Zealand. Email: andre@cs.auckland.ac.nz.

‡S. Rubin, Department of Computer Science, University of Auckland, New Zealand. Email: rubin@cs.auckland.ac.nz. S. Rubin is supported by a New Zealand Science and Technology Post-Doctoral Fellowship.

§F. Stephan, School of Computing and Department of Mathematics, National University of Singapore, Republic of Singapore. Email: fstephan@comp.nus.edu.sg. F. Stephan previously worked at National ICT Australia which is funded by the Australian Government's Department of Communications, Information Technology and the Arts and the Australian Research Council through Backing Australia's Ability and the ICT Centre of Excellence Program. Currently, F. Stephan is supported in part by NUS grant number R252-000-212-112.

extend the framework of finite model theory to infinite models that have finite presentations.

Automatic structures are (usually) infinite relational structures whose domain and atomic relations can be recognised by finite automata operating synchronously on their input. Consequently, automatic structures have finite presentations and are closed under first order interpretability (as well as some of extensions of first order logic). Moreover, the model checking problem for automatic structures is decidable. Hence automatic structures and tools developed for their study are well suited to these fields of computer science, see for instance [2].

From a computability and logical point of view, automatic structures are used to provide generic examples of structures with decidable theories, to investigate the relationship between automata and definability, and to refine the ideas and approaches in the theory of computable structures. This paper investigates the problem of characterising automatic structures in algebraic, model theoretic or logical terms.

Specifically this paper addresses two foundational problems in the theory of automatic structures. The first is that of providing *structure theorems* for classes of automatic structures. Fix a class $\mathcal{C}$ of structures (over a given signature), closed under isomorphism. For instance, $\mathcal{C}$ may be the class of groups $(G, \cdot)$ or the class of linear orders $(L, \leq)$. A structure theorem should be able to distinguish whether a given member of $\mathcal{C}$ has an automatic presentation or not (a special case of this is telling whether a given structure is automatically presentable or not). This usually concerns the interactions between the combinatorics of the finite automata presenting the structures and properties of the structures themselves. The second problem, which is related to the first, is the complexity of the *isomorphism problem* for classes of automatic structures. Namely, fix a class of automatic structures $\mathcal{C}$. The isomorphism problem asks, given automatic presentations of two structures from $\mathcal{C}$, are the structures isomorphic?

With regard to the first problem, we provide new techniques for proving that some foundational structures in computer science and mathematics do not have automatic presentations. For example, we show that the Fraïssé limits of many classes of finite structures, such as finite partial orders or finite graphs, do not have automatic presentations. This shows that the infinite random graph and random partial order do not have automatic presentations. The idea is that the finite amount of memory intrinsic to the finite automata presenting the structure can be used to extract algebraic and model theoretic properties (invariants) of the structure, and so used to classify such structures up to isomorphism. This line of research has indeed been successful in investigating automatic ordinals, linear orders, trees and Boolean algebras. For example there is a full structure theorem for the automatically presentable ordinals; namely, they are those strictly less than ω^ω [7]. There are also partial structure theorems saying that automatic linear orders and automatic trees have finite Cantor-Bendixson rank [13]. In this paper we provide a structure theorem for the (infinite) automatic Boolean algebras; namely, they are those isomorphic to finite products of the

Boolean algebra of finite and co-finite subsets of $\mathbb{N}$.

With regard to the second problem, it is not surprising that the isomorphism problem for the class of all automatic structures in undecidable [5]. The reason for the undecidability is that the configuration space of a Turing machine, considered as a graph, is an automatic structure. Moreover, the reachability problem in the configuration space is undecidable. Thus with some extra work, as in [4] or [11], one can reduce the reachability problem to the isomorphism problem for automatic structures. In addition, the isomorphism problem for automatic ordinals [13] and Boolean algebras (Corollary 3.5) is decidable. For equivalence structures it is in Π^0_1 (though not more is known), and for configuration spaces of Turing machines is Π^0_3-complete [16].

Hence it is somewhat unexpected that the complexity of the isomorphism problem for the class of automatic structures is Σ^1_1-complete. The Σ^1_1-completeness is proved by reducing the isomorphism problem for computable trees, known to be Σ^1_1-complete [9], to the isomorphism problem for automatic structures.

The two problems are related in the following way. If one has a 'nice' structure theorem for a class $\mathcal{C}$ of automatic structures, then one expects that the isomorphism problem for $\mathcal{C}$ be computationally 'reasonable'. For instance, as corollaries of the structure theorems for automatic ordinals and for automatic Boolean algebras, one obtains that their corresponding isomorphism problems are decidable. In contrast, the Σ^1_1-completeness of the isomorphism problem of the class of all automatic structures tells us that the language of first order arithmetic is not powerful enough to give a structure theorem for the class of all automatic structures. In other words we should not expect a 'nice' structure theorem for the class of all automatic structures.

Here is an outline of the rest of the paper. The next section is a brief introduction to the basic definitions. Section 3 provides some counting techniques sufficient to prove non-automaticity of many classical structures such as fields, integral domains and Boolean algebras. Section 4 provides a technique that is used to show non-automaticity of several structures, such as the infinite random graph and the random partial order. The last section is devoted to proving that the isomorphism problem for automatic structures is Σ^1_1-complete.

Finally, we note that automatic structures can be generalised in several directions: for instance, by using finite automata on infinite strings [4, 5], finite ranked trees [4, 1], or finite unranked trees [14]. Although this paper only deals with the finite word case, some of the ideas presented here should give rise to a better understanding of the generalisations. Indeed, Delhommé, who independently proved that the random graph has no automatic presentation (using a similar technique as the one presented here) has extended the technique to tree-automatic structures [7].

2 Preliminaries

A thorough introduction to automatic structures can be found in [4] and [12]. We assume familiarity with the basics of finite automata theory though to fix notation the necessary definitions are included. 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 $\sigma_1\sigma_2 \dots \sigma_n$ $(\sigma_i \in \Sigma)$ is a sequence of states, say $q_0, q_1, \dots, q_n$, such that $q_0 = \iota$ and $(q_i, \sigma_{i+1}, q_{i+1}) \in \Delta$ for all $i \in \{0, 1, \dots, n-1\}$. 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 the language accepted by a finite automaton $\mathcal{A}$.

The following definitions extends recognisability to relations of arity n, called *synchronous n–tape automata*. A synchronous n–tape automaton can be thought of as a one-way Turing machine with n input tapes [8]. 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 definition.

Definition 2.1 *Write $\Sigma_\diamond$ for $\Sigma \cup \{\diamondsuit\}$ where $\diamondsuit$ is a symbol not in Σ. The* convolution of a tuple $(w_1, \cdots, w_n) \in \Sigma^{\star n}$ *is the string $\otimes(w_1, \cdots, w_n)$ of length* $\max_i |w_i|$ *over alphabet $(\Sigma_\diamond)^n$ defined as follows. Its k'th symbol is $(\sigma_1, \dots, \sigma_n)$ where σ_i is the k'th symbol of w_i if $k \le |w_i|$ and $\diamond$ otherwise.*

The convolution of a relation $R \subset \Sigma^{\star n}$ *is the relation $\otimes R \subset (\Sigma_\diamond)^{n\star}$ formed as the set of convolutions of all the tuples in R. That is $\otimes R = \{\otimes w \mid w \in R\}$.*

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 *(in short FA recognisable) or* regular *if its convolution $\otimes R$ is recognisable by an n–tape automaton.*

We now relate n–tape automata to structures. A *structure* $\mathcal{A}$ consists of a countable set A called the *domain* and some relations and operations on A. We may assume that $\mathcal{A}$ only contains relational predicates as the operations can be replaced with their graphs. We write $\mathcal{A} = (A, R_1^A, \dots, R_k^A, \dots)$ where R_i^A is an n_i–ary relation on $\mathcal{A}$. The relation R_i are sometimes called basic or atomic relations. We assume that the function $i \to n_i$ is always a computable one.

Definition 2.3 *A structure $\mathcal{A}$ is* automatic *over Σ if its domain $A \subset \Sigma^\star$ is finite automata recognisable, and there is an algorithm that for each i produces a finite automaton recognising the relation $R_i^A \subset (\Sigma^\star)^{n_i}$. A structure is called* automatic *if it is automatic over some alphabet. If $\mathcal{B}$ is isomorphic to an automatic structure $\mathcal{A}$, then call $\mathcal{A}$ an* automatic presentation *of $\mathcal{B}$ and say that $\mathcal{B}$ is called* automatically presentable (*over* Σ).

An example of an automatic structure is the word structure $(\{0,1\}^\star, L, R, E, \preceq)$, where for all $x, y \in \{0,1\}^\star$, $L(x) = x0$, $R(x) = x1$, $E(x,y)$ iff $|x| = |y|$, and $\preceq$ is the lexicographical order. The configuration graph of any Turing machine is another example of an automatic structure. Examples of automatically presentable structures are $(\mathbb{N}, +)$, $(\mathbb{N}, \leq)$, $(\mathbb{N}, S)$, the group $(\mathbb{Z}, +)$, the order on the rationals $(Q, \leq)$, and the Boolean algebra of finite or co-finite subsets of $\mathbb{N}$. Note that every finite structure is automatically presentable. We use the following important theorem without reference.

Theorem 2.4 [12] *Let $\mathcal{A}$ be an automatic structure. There exists an algorithm that from a first order definition (with possible use of the additional quantifier 'there exists infinitely many') in $\mathcal{A}$ of a relation R produces an automaton recognising R.*

3 Proving Non-Automaticity via Counting

The first technique for proving non-automaticity was presented in [12] and later generalised in [4]. The technique is based on a pumping argument and exhibits the interplay between finitely generated (sub) algebras and finite automata. We briefly recall the technique for completeness.

A relation $R \subset (\Sigma^\star)^n$ is called *locally finite* if there exists k, l with $k + l = n$ so that for every $\bar{a}$ (of size k) there are at most a finite number of $\bar{b}$ (of size l) such that $(\bar{a}, \bar{b}) \in R$. We write $R \subset (\Sigma^\star)^{k+l}$. For $\bar{b} = (b_1, \cdots, b_m)$, write $b \in \bar{b}$ to mean $b = b_i$ for some i.

We start with the following elementary but important proposition.

Proposition 3.1 *Let $R \subset (\Sigma^\star)^{k+l}$ be locally finite, with k and l as in the definition above. Suppose further that R is a regular relation, and that the automaton for $\otimes(R)$ has p states. Then*

$$\max\{|y| \mid y \in \overline{y}\} - \max\{|x| \mid x \in \overline{x}\} \leq p$$

for every $(\overline{x}, \overline{y}) \in R$ where $\overline{x}$ has k elements and $\overline{y}$ has l elements.

Proof. Fix $(\overline{x}, \overline{y}) \in R$ and say $x' \in \overline{x}$ has length $\max\{|x| \mid x \in \overline{x}\}$ and say $y' \in \overline{y}$ has length $\max\{|y| \mid y \in \overline{y}\}$. So $|y'| - |x'| > p$ implies that we can pump the string $\otimes(\overline{x}, \overline{y})$ between positions $|x'|$ and $|y'|$. Then either the automaton for $\otimes(R)$ accepts a string that is not in $\otimes((\Sigma^\star)^n)$ (because one of the components contains a subword of the form $\diamond\Sigma$) or otherwise it accepts strings of the form $\otimes(\overline{x}, \overline{z})$ for infinitely many $\overline{z}$, contradicting that R is locally finite. $\Box$

The typical application of this proposition is to prove that certain structures do not have automatic presentations. Assume $\mathcal{A}$ is an automatic structure in which each atomic relation R_i is a graph of a function f_i, $i = 1, \dots, m$. Let $a_1, a_2, \dots$ be a sequence of some elements of A such that the relation $\{(a_i, a_j) \mid j = i+1\}$ is regular. Consider the sequence $G_1 = \{a_1\}$, $G_{n+1} = G_n \cup \{a_{n+1}\} \cup \{f_i(\bar{a}) \mid \bar{a} \in G_n, i = 1, \dots, m\}$. By the proposition there is a constant p such that the length of all elements in G_n is bounded by $p \cdot n$. Therefore the number of elements in G_n is bounded by $2^{O(n)}$. Some combinatorial reasoning combined with this observation can now be applied to provide examples of structures with no automatic presentations, see [4] and [12]. For example, the following structures have no automatic presentation:

1. The free group on $k > 1$ generators;
2. The structure $(\mathbb{N}, |)$;
3. The structure $(\mathbb{N}, p)$, where $p : \mathbb{N}^2 \to \mathbb{N}$ is a bijection;
4. The term algebra generated by finitely many constants with at least one non-unary atomic operation[1].

Note that each of these structures has a decidable first order theory.

In the next sections we provide other more intricate techniques for showing that particular structures do not have automatic presentations. We then apply those techniques to give a characterisation of Boolean algebras that have automatic presentations. We also prove that $(\mathbb{Q}^+, \times)$ has no automatic presentation and show that no infinite integral domain (in particular no infinite field) has an automatic presentation. We also study automaticity of some Fraïssé limits.

First we introduce a very useful property true of every automatic monoid $(M, \times)$.

Lemma 3.2 *For each $s_1, \dots, s_m \in M$, $|\prod_i s_i| \le \max_i |s_i| + k\lceil \log m \rceil$, where k is the number of states in the automaton recognising the graph of $\times$.*

Proof. Here logarithm is to base 2 and $\lceil \log n \rceil$ is the least i such that $2^i \ge n$. We use induction on m. For $m = 1$, the inequality becomes $|s_1| \le |s_1|$. If $m > 1$ let $m = u + v$ where $u = \lfloor m/2 \rfloor$. Apply Proposition 3.1 to the graph of the monoid operation $\times$ and elements $x = \prod_{i=1}^{u} s_i$ and $y = \prod_{i=u+1}^{m} s_i$. Then, by induction, $|\prod_i s_i| \le k + \max(|x|, |y|)$ which is equal to

$$k + \max(\max_{1 \le i \le u} |s_i| + k\lceil \log u \rceil, \max_{u+1 \le i \le m} |s_i| + k\lceil \log v \rceil),$$

which is at most $\max_i |s_i| + k\lceil \log m \rceil$, since $1 + \max(\lceil \log u \rceil, \lceil \log v \rceil) \le \lceil \log m \rceil$. $\Box$

[1] Thus, elements of the term algebra are all the ground terms, and the operations are defined in a natural way: the value of a function f of arity n from the language on ground terms $g_1, \dots, g_n$ is $f(g_1, \dots, g_n)$.

3.1 Automatic Boolean algebras

All finite Boolean algebras are automatic. Thus, in this section we deal with infinite countable Boolean algebras only. Our goal in this section is to give a full characterisation of infinite automatic Boolean algebras. Our characterisation can then be applied to show that the isomorphism problem for automatic Boolean algebras is decidable. Compare this with the result from computable algebra that the isomorphism problem for computable Boolean algebras is Σ_1^1-complete [9].

Recall that a Boolean algebra $\mathcal{B} = (B, \cup, \cap, \setminus, \mathbf{0}, \mathbf{1})$ is a structure, where $\cap$ and $\cup$ and $\setminus$ operations satisfy all the basic properties of the set-theoretic intersection, union, and complementation operations; In $\mathcal{B}$ the relation $a \subseteq b \iff a \cap b = a$ is a partial order in which $\mathbf{0}$ is the smallest element, and $\mathbf{1}$ is the greatest element. The complement of an element $b \in B$ is $\mathbf{1} \setminus b$ and is denoted by $\bar{b}$.

A linearly ordered set determines a Boolean algebra in a natural way described as follows. Let $\mathcal{L} = (L, \leq)$ be a linearly ordered set. An interval is a subset of L of the form $[a, b) = \{x \mid a \leq x < b\}$, where $a, b \in L \cup \{\infty\}$. The **interval algebra** denoted by $\mathcal{B}_{\mathcal{L}}$ is the collection of all finite unions of intervals of $\mathcal{L}$, with the usual set-theoretic operations of intersection, union and complementation. Every interval algebra is a Boolean algebra. Moreover for every countable Boolean algebra $\mathcal{A}$ there exists an interval algebra $\mathcal{B}_{\mathcal{L}}$ isomorphic to $\mathcal{A}$. We write $\mathcal{L}_1 \times \mathcal{L}_2$ for the ordered sum $\sum_{l \in \mathcal{L}_2} \mathcal{L}_1$.

Lemma 3.3 *The interval Boolean algebras $\mathcal{B}_{\omega \times i}$, where i is positive integer, all have automatic presentations.*

Proof. The Boolean algebra $\mathcal{B}_\omega$ has an automatic presentation. Indeed, every element X of $\mathcal{B}_\omega$ can be represented by a string that codes the characteristic function of X. For example, the element $[1, 3) \cup [6, 10)$ can be represented by the string 0#0110001111 while $\mathbb{N} \setminus [3, 4)$ can be represented by the string 1#0001. The boolean operations under this representation are regular, hence this is an automatic presentation of $\mathcal{B}_\omega$. Now, $\mathcal{B}_{\omega \times i}$ is isomorphic to the Cartesian product of i copies of $\mathcal{B}_\omega$. Automatic structures are closed under the Cartesian product, and this completes the proof. $\square$

An **atom** in a Boolean algebra is a non-zero element a such that for every $b \leq a$ we have $a = b$ or $b = \mathbf{0}$. Assume that $\mathcal{B}$ is an automatic Boolean algebra not isomorphic to any of the algebras $\mathcal{B}_{\omega \times i}$. Call two elements $a, b \in B$ ***F*-equivalent** if the element $(a \cap \bar{b}) \cup (b \cap \bar{a})$ is a union of finitely many atoms. Factorise $\mathcal{B}$ with respect to the equivalence relation. Denote the factor algebra by $\mathcal{B}/F$. Due to the assumption on $\mathcal{B}$ the algebra $\mathcal{B}/F$ is not finite. Call x in $\mathcal{B}$ **large** if its image in $\mathcal{B}/F$ is not a finite union of atoms or $\mathbf{0}$. For example the element $\mathbf{1}$ is large in $\mathcal{B}$ because $\mathcal{B}$ is not isomorphic to $\mathcal{B}_{\omega \times i}$. Call an element x in $\mathcal{B}$ **infinite** if there are infinitely many elements below it. Say that x **splits** y, for $x, y \in B$, if $x \cap y \neq \mathbf{0}$ and $\bar{x} \cap y \neq \mathbf{0}$. For every large element $l \in B$ there exists an element $x \in B$ that splits l such that $x \cap l$ is large and $\bar{x} \cap l$ is infinite.

Also for every infinite element $i \in B$ there exists an element $x \in B$ that splits i such that either $x \cap i$ or $\bar{x} \cap i$ is infinite.

We are now ready to prove the following theorem characterising infinite automatic Boolean algebras.

Theorem 3.4 *An infinite Boolean algebra has an automatic presentation if and only if it is isomorphic to $\mathcal{B}_{\omega \times i}$ for some positive $i \in \mathbb{N}$.*

Proof. We first construct a sequence T_n of trees and elements $a_\sigma \in B$ corresponding to elements $\sigma \in T_n$ as follows. The tree T_n will be a set of binary strings closed under prefixes. Therefore it suffices to define leaves of T_n. Initially, we set $T_0 = \{\lambda\}$ and $a_\lambda = \mathbf{1}$. Assume that T_n has been constructed. By induction hypothesis we may assume that the leaves of T_n satisfy the following properties: (1) There exists at least one leaf σ such that a_σ is large in $\mathcal{B}$. Call the element a_σ leading in T_n. (2) There exist n leaves $\sigma_1, \ldots, \sigma_n$ such that each a_{σ_i} is an infinite element in $\mathcal{B}$. Call these elements sub-leading elements. (3) The number of leaves in T_n is greater than or equal to $n \cdot (n+1)/2$. (4) For every pair of leaves x, y of T_n it holds that $x \cap y = \mathbf{0}$.

For each leaf $\sigma \in T_n$ proceed as follows:

1. If σ is leading then find the first length lexicographical b that splits a_σ such that both $a_\sigma \cap b$ and $a_\sigma \cap \bar{b}$ are infinite and one of them is large. Put $\sigma 0$ and $\sigma 1$ into T_{n+1}. Set $a_{\sigma 0} = a_\sigma \cap b$ and $a_{\sigma 1} = a_\sigma \cap \bar{b}$.

2. If σ is a sub-leading then find the first length lexicographical b that splits a_σ such that one of $a_\sigma \cap b$ or $a_\sigma \cap \bar{b}$ is infinite. Put $\sigma 0$ and $\sigma 1$ into T_{n+1}. Set $a_{\sigma 0} = a_\sigma \cap b$ and $a_{\sigma 1} = a_\sigma \cap \bar{b}$.

Thus, we have constructed the tree T_{n+1} and elements a_σ corresponding to the leaves of the tree. Note that the inductive hypothesis holds for T_{n+1}. This completes the definition of the trees.

Lemma 3.2 is now used a number of times to justify the following steps. There exists a constant c_1 such that $|a_{\sigma\epsilon}| \leq |a_\sigma| + c_1$ for all defined elements a_σ . Now for every n consider the set $X_n = \{a_\sigma \mid \sigma \text{ is a leaf of } T_n\}$. There exists a constant c_2 such that for all $x \in X_n$ we have $|x| \leq c_2 \cdot n$. Therefore $X_n \subset \Sigma^{c_2 \cdot n}$ and the number of leaves in T_n is greater than or equal to $n \cdot (n+1)/2$. Now for every pair of elements a, b in X_n we have $a \cap b = \mathbf{0}$. Therefore the number of elements of the Boolean algebra generated by the elements in X_n is $2^{|X_n|}$. Now let $Y = \{b_1, \ldots, b_k\} \subset X_n$. Consider the element $\cup Y = b_1 \cup \ldots \cup b_k$. By Lemma 3.2 applied to the binary operation $\cup$ there exists a constant c_3 such that $|\cup Y| \leq c_3 \cdot n$. This gives us a contradiction because the number of elements generated by elements of X_n clearly exceeds the cardinality of $\Sigma^{O(n)}$. $\Box$

Corollary 3.5 *It is decidable whether two automatic Boolean algebras are isomorphic.*

Proof. Every automatic Boolean algebra is isomorphic to the Cartesian product of i copies of B_ω, the Boolean algebra of finite and co-finite subsets of $\mathbb{N}$. This i can be obtained effectively: Given an FA-presentation of a structure A in the signature of Boolean algebras, one can decide if A is a Boolean algebra, and if so compute the largest i such that A models

> "there are i disjoint elements each with infinitely many atoms below."

Thus the isomorphism problem is decidable. $\Box$

3.2 Commutative monoids and Abelian groups

Note that for groups and monoids the term 'automatic' is used in a different way [6]. So to avoid confusion we say such a structure is 'FA presented' instead of saying it is 'automatic', and it is 'FA presentable' instead of 'automatically presentable'.

We prove that $(\mathbb{Q}^+, \times)$, or equivalently, the free Abelian group of rank ω, is not FA presentable.

Theorem 3.6 *Let $(M, \times)$ be a monoid containing $(\mathbb{N}, \times)$ as a submonoid. Then $(M, \times)$ is not FA presentable.*

Proof. Assume for a contradiction that an FA presentation of M is given. Let $a_0, a_1, \ldots$ be the prime numbers, viewed as elements of M, and listed in length-lexicographical order (with respect to this presentation of M). Let r_n be such that $a_0, \ldots, a_{r_n-1}$ are the primes of length at most n. Let

$$F_n = \{\textstyle\prod_{i\,:\,0 \leq i < r_n} a_i^{\beta_i} : 0 \leq \beta_i < 2^n\}.$$

By Lemma 3.2, each term $a_i^{\beta_i}$ has length at most $n + k \log \beta_i \leq n(1+k)$. Again by the lemma, each product has length at most $n(1+k) + k \log r_n$. Since all the products are distinct,

$$2^{nr_n} \leq |F_n| \leq |\Sigma|^{(1+k)n + k \log r_n}.$$

Thus $nr_n \leq \log|\Sigma|[(1+k)n + k \log r_n]$ and $r_n \in O(\log r_n)/n$, a contradiction because r_n goes to infinity. $\Box$

In [15] a stronger form is proved: if $(\mathbb{N}, +)^r$ is a submonoid of M, then $r \leq \log|\Sigma|(k+1)$, where Σ is the alphabet and k the number of states needed to recognize the graph of the operation of M.

Corollary 3.7 $(\mathbb{Q}^+, \times)$ *is not FA presentable.*

For a prime p, let $\mathbb{Z}[1/p]$ be the additive group of rationals of the form z/p^m, z an integer, $m \in \mathbb{N}$, and let $\mathbb{Z}_{p^\infty}$ be the Prüfer group $\mathbb{Z}[1/p]/\mathbb{Z}$. Using representations to base p, it is easy to give FA presentations of these groups. Hence finite direct sums of those groups are also FA presentable. The proof of the following uses similar methods to the ones of Theorem 3.6.

Theorem 3.8 *Let $A^{(\omega)}$ denote the direct sum of infinitely many copies of the group A. The infinite direct sums $\mathbb{Z}[1/p]^{(\omega)}$ and $\mathbb{Z}_{p^\infty}^{(\omega)}$ are not FA presentable.*

Proof. The proof is similar to the proof of Theorem 3.6. Suppose there is an FA presentation, and the i-th copy of the group in question is generated by all elements of the form a_i/p^m, $i, m \in \mathbb{N}$, where the elements $a_0, a_1, \dots$ are listed in length-lexicographical order. Define r_n as before, and consider sums $\sum_{i<r_n} \beta_i a_i/p^n$, where $0 \le \beta_i < p^n$. By Proposition 3.1, the definable operation $x \mapsto x/p$ increases the length by at most a constant. So, using Lemma 3.2 each term $\beta_i a_i/p^n$ has length at most $n + cn + kn \log p$ for appropriate $c \in \mathbb{N}$. Thus each sum has length at most $c'n + k \log r_n$. As there are p^{nr_n} distinct sums, this yields a contradiction as before. $\Box$

3.3 Integral domains

In our next result we prove that no infinite integral domain is FA presentable. The following definition and lemma will be used in the next section as well.

Definition 3.9 *Suppose $\mathcal{D}$ is a structure over alphabet Σ. Write $D^{\le n}$ for $D \cap \Sigma^{\le n}$; that is the elements of D of length at most n. Write $P_n(D)$ for $\{x \in \Sigma^n \mid \exists z \in \Sigma^\star \wedge xz \in D\}$, namely all prefixes of length n of all words in the domain.*

Lemma 3.10 *If $D \subset \Sigma^\star$ is a regular language then*

1. *$|P_n(D)| \in O(|D^{\le n}|)$ and*
2. *$|D^{\le n+k}| \in \Theta(|D^{\le n}|)$ for every constant $k \in \mathbb{N}$.*

Proof. Suppose the automaton recognising D has c states. Then for $x \in P_n(D)$ there exists $z \in \Sigma^\star$ with $|z| \le c$ such that $xz \in D$ (†). If $n \ge c$ then $|P_n(D)| \le |\Sigma|^c \times |P_{n-c}(D)|$ since the map associating $x \in P_n(D)$ with the word consisting of the first $n - c$ letters of x, is $|\Sigma|^c$-to-one. But by using (†) we see that $|P_{n-c}(D)| \le |D^{\le n}|$. So $|P_n(D)| \le |\Sigma|^c \times |D^{\le n}|$ as required for the first part.

Fix $k \in \mathbb{N}$. The mapping associating $x \in D^{\le n+k}$ to the prefix of x of length n is $|\Sigma|^k$-to-one. Hence

$$|D^{\le n+k}| \le |\Sigma|^k \times |P_n(D)| \le |\Sigma|^k \times |\Sigma|^c \times |D^{\le n}|.$$

Since $D^{\le n} \subset D^{\le n+k}$ one has that $|D^{\le n}| \le |D^{\le n+k}|$ and $|D^{\le n+k}| \in O(|D^{\le n}|)$. This completes the proof of the lemma. $\Box$

Recall that an integral domain $(D, +, \cdot, 0, 1)$ is a commutative ring with identity such that $x \cdot y = 0$ only if $x = 0$ or $y = 0$. For example every field is an integral domain.

Theorem 3.11 *No infinite integral domain is FA presentable.*

Proof. Suppose that $(D, +, \cdot, 0, 1)$ is an infinite automatic integral domain. For each $n \in \mathbb{N}$ recall that $D^{\leq n} = \{u \in D \mid |u| \leq n\}$. We claim that there exists an x in D such that for all $a, b, a', b' \in D^{\leq n}$ the condition $a \cdot x + b = a' \cdot x + b'$ implies that $a = a'$ and $b = b'$. We say such x **separates** $D^{\leq n}$. Indeed, assume that such an x does not exist. Then for each $x \in D$ there exist $a, b, a', b' \in D^{\leq n}$ such that $a \cdot x + b = a' \cdot x + b'$ but $(a, b) \neq (a', b')$. Hence, since $D^{\leq n}$ is finite, there exist $a, b, a', b' \in D^{\leq n}$ such that $a \cdot x + b = a' \cdot x + b'$ but $(a, b) \neq (a', b')$ for infinitely many x. Thus, for infinitely many x we have $(a - a') \cdot x = b' - b$. But $a \neq a'$ for otherwise also $b = b'$, contrary to assumption. Also there exist distinct x_1 and x_2 such that $(a - a') \cdot x_1 = (a - a') \cdot x_2$. Since D is an integral domain we conclude that $x_1 = x_2$ which is a contradiction.

For each $D^{\leq n}$ we can select the length-lexicographically first x_n separating $D^{\leq n}$. Now the set $E_n = \{y \mid \exists a, b \in D^{\leq n}\,[y = ax_n + b]\}$ has at least $|D^{\leq n}|^2$ many elements. However by Proposition 3.1 there exists a constant C such that $E_n \subset D^{\leq n+C}$, and by Lemma 3.10 the number of elements in $D^{\leq n+C}$ is in $O(|D^{\leq n}|)$. Thus, we have a contradiction. The theorem is proved. $\square$

Corollary 3.12 *No infinite field is FA presentable.*

4 Non-automaticity of some Fraïssé Limits

We briefly recall the definition of Fraïssé limit; see for instance [10]. In this section we restrict consideration to relational structures of finite signature, although the definition can be extended to signatures with function symbols. Let K be a class of finite structures closed under isomorphism. We say that K has the **hereditary property (HP)** if for $\mathcal{A} \in K$ every substructure of $\mathcal{A}$ is also in K. We say that K has the **joint embedding property (JEP)** if for all $\mathcal{A}, \mathcal{B} \in K$ there exists $\mathcal{C} \in K$ such that $\mathcal{A}$ and $\mathcal{B}$ are both embeddable into $\mathcal{C}$. We say that K is the **age** of a structure $\mathcal{D}$ if K coincides with the class of all finite substructures of $\mathcal{D}$. It clear that the age of a structure has HP and JEP. In fact, it is not hard to prove that a class K has HP and JEP if and only if K coincides with the class of all finite substructure of some structure. However, note that non-isomorphic structures may have the same age. The following property guarantees that a class K with HP and JEP defines a unique structure up to isomorphism:

Definition 4.1 *A class K of finite structures has* **amalgamation property (AP)** *if for $\mathcal{A}, \mathcal{B}, \mathcal{C} \in K$ with embeddings $e : \mathcal{A} \to \mathcal{B}$ and $f : \mathcal{A} \to \mathcal{C}$ there are $\mathcal{D} \in K$ and embeddings $g : \mathcal{B} \to \mathcal{D}$ and $h : \mathcal{C} \to \mathcal{D}$ such that $ge = hf$.*

Below we cite a classical result in model theory due to Fraïssé . For this we mention that a structure $\mathcal{A}$ is called **ultra-homogeneous** if every isomorphism between finite substructures of $\mathcal{A}$ can be extended to an automorphism of $\mathcal{A}$. For the proof the reader may consult [10, Theorem 7.1.2].

Theorem 4.2 *Let K be a nonempty class of finite structures which has HP, JEP, and AP. Then there exists a up to isomorphism unique countable structure*

$\mathcal{A}$, called the **Fraïssé limit of** *K, such that $\mathcal{A}$ is ultra-homogeneous and K is the age of $\mathcal{A}$.*

Now we can apply the theorem to obtain several examples of structures. In each of these examples the classes K have HP, JEP, and AP.

Example 4.3 *Let K be the class of all finite linear orders. The Fraïssé limit of K is isomorphic to the ordering of rationals.*

Example 4.4 *Let K be the class of all finite linear orders with one unary predicate. The Fraïssé limit of K is isomorphic $(Q; \leq, U)$, where $\leq$ is the linear order of rationals Q, and U is a dense and co-dense subset of Q.*

Example 4.5 *Let K be the class of all finite graphs. The Fraïssé limit of K is known as the* **random graph**. *The following is an algebraic property that gives a characterisation (of the isomorphism type) of the random graph $\mathcal{R} = (V, E)$ (see [10]). For every disjoint partition X_1, X_2 of every finite set Y of vertices there exists a vertex y such that for all $x_1 \in X_1$ and $x_2 \in X_2$ we have $(y, x_1) \in E$ and $(y, x_2) \notin E$. An explicit description of a random graph is the following. The set V of vertices is $\mathbb{N}$ and $(n, m) \in E$ if in the binary representation of n, the term 2^m has coefficient 1.*

Example 4.6 *Let K be the class of all finite structures of a given signature L. The Fraïssé limit of K is known as the* **random L-structure**.

Example 4.7 *Denote by $\mathcal{K}_p$ the complete graph (every pair of vertices are connected by an edge) on p vertices. For $p \geq 3$, consider the class of all finite graphs which do not contain K_p as a subgraph. The Fraïssé limit of K is known as the* **random $\mathcal{K}_p$-free graph**. *It has the following property. For every finite K_{p-1}-free subgraph, say with domain Y, and every disjoint partition X_1, X_2 of Y, there exists a vertex x that is edge connected to every vertex in X_1 and no vertex in X_2.*

Recall that an anti-chain in a partial order is a set of pairwise incomparable elements. A chain in a partial order is a set in which every pair of elements are comparable.

Example 4.8 *Let K be the class of all finite partially ordered set. The Fraïssé limit $\mathcal{U}$ of K is known as the* **random partial order**. *The following is an algebraic characterisation (of the isomorphism type) of the random partial order $\mathcal{U} = (U, \leq)$ (see [10]).*

1. *If Z is a* finite *anti-chain of $\mathcal{U}$ and X and Y partition Z then there exists an element $z \in U$ such that for every $x \in X$, $z > x$ and for every $y \in Y$, element z is not comparable with y.*

2. *If Z is a* finite *chain of $\mathcal{U}$ with least element x and largest element y then there exists an element $z \in U$ such that $z > x$ and $z < y$ and z is not comparable with every $v \in X \setminus \{x, y\}$.*

Example 4.9 *Let K be the class of all finite Boolean algebras. The Fraïssé limit of K is isomorphic to the atomless Boolean algebra. This is the Boolean algebra that satisfies the following property. For every non-zero element x there exists a nonzero y strictly below x (that is $y < x$). By Theorem 3.4 this Fraïssé limit has no automatic presentation.*

Example 4.10 *Let K be the class of all finite Abelian p-groups. The Fraïssé limit of K is isomorphic to $G = (\mathbb{Z}_{p^\infty})^\omega$.*

Note that G has no FA-presentation by Theorem 3.8.
Proof. The class K has HP, JEP and AP, so the Fraïssé limit exists. To show this Fraïssé limit is isomorphic to G, by [10, Lemma 7.1.4] it suffices to show that the age of G is K (clear) and that G is weakly homogeneous. To do so, suppose $A \subseteq B$ are in K. We have to show that each embedding of A into G extends to an embedding of B. We may assume that $|B : A| = p$. Since B is a direct product of cyclic groups whose order is a power of p, either $B = A \times \mathbb{Z}_p$ or there is $x \in A$ such that x is not divisible by p in A, and $py = x$ for some $y \in B$. In either case we can extend the embedding. □

Below are examples of Fraïssé limits that have automatic presentations.

Example 4.11 *The linear order of rational numbers has an automatic presentation. In fact it is straightforward to check that $(\{0,1\}^\star \cdot 1, \preceq_{lex})$ is an automatic presentation of $(Q, \leq)$.*

Example 4.12 *Let $U = \{u \mid u \in \{0,1\}^\star \cdot 1,\ |u| \text{ is even}\}$. The structure $(\{0,1\}^\star \cdot 1; \preceq_{lex}, U)$ is the Fraïssé limit for the class K of all finite linear orders with one unary predicate.*

We now provide some methods for proving non-automaticity of structures. These methods are then applied to prove that some Fraïssé limits do not have automatic presentations.

Let $\mathcal{A}$ be an automatic structure over the alphabet Σ. Recall $A^{\leq n} = \{v \in A \mid |v| \leq n\}$. Let $\Phi(x, y)$ be a two variable formula in the language of this structure. We do not exclude that $\Phi(x, y)$ has a finite number of parameters from the domain of the structure. Now for each $y \in A$ and $n \in \mathbb{N}$ we define the following function $c^\Phi_{n,y} : A^{\leq n} \to \{0, 1\}$:

$$c^\Phi_{n,y}(x) = \begin{cases} 1 & \text{if} \quad \mathcal{A} \models \Phi(x, y); \\ 0 & \text{if} \quad \mathcal{A} \models \neg\Phi(x, y). \end{cases}$$

We may drop the superscript Φ if there is no danger of ambiguity. In the next theorem we count the number of functions $c^\Phi_{n,y}$ using the fact that $\mathcal{A}$ is an automatic structure. We will use this as a criterion for proving that a given structure is not automatically presentable.

Theorem 4.13 *Let $\mathcal{A}$ be an automatic structure and $\Phi(x, y)$ a first order formula (possibly with parameters) over the language of $\mathcal{A}$. Then the number of functions $c^\Phi_{n,y}$ is in $O(|A^{\leq n}|)$.*

Proof. It is sufficient to prove that there is a constant k so that the number of functions of the form $c_{n,y}$ with $y \in A \cap \Sigma^{>n}$ is at most $k(|A^{\leq n}|)$; this is because the y's in $A^{\leq n}$ can supply at most $|A^{\leq n}|$ many additional functions $c_{n,y}$.

Let (Q, ι, ρ, F) be a deterministic automaton recognising the relation $\otimes\Phi = \{\otimes(x,y) \mid \mathcal{A} \models \Phi(x,y)\}$. Fix $n \in \mathbb{N}$. We will associate with each $c_{n,y}$, where $y \in A \cap \Sigma^{>n}$, two pieces of information; namely, a function $J_y : A^{\leq n} \to Q$ and a set $Q_y \subset Q$ as follows. Let $\otimes(x,y) = \sigma_1\sigma_2 \cdots \sigma_k$, where $x \in A^{\leq n}$ and $\sigma_i \in (\Sigma_\diamond)^2$. Then define $J_y(x) := \rho(\iota, \sigma_1 \cdots \sigma_n)$. Define $Q_y \subset Q$ as those states $s \in Q$ such that $\rho(s, \sigma_{n+1} \cdots \sigma_k) \in F$. Note that $\sigma_{n+1} \cdots \sigma_k = \otimes(\lambda, z)$, for some $z \in \Sigma^*$, and is independent of x.

We claim that if $(J_y, Q_y) = (J_{y'}, Q_{y'})$ then $c_{n,y} = c_{n,y'}$. So suppose that $(J_y, Q_y) = (J_{y'}, Q_{y'})$, and let $x \in A^{\leq n}$ be given. Say $\otimes(x,y) = \sigma_1 \cdots \sigma_k$, and $\otimes(x,y') = \delta_1 \cdots \delta_l$. Then

$$\begin{aligned}
\mathcal{A} \models \phi(x,y) \quad &\Longleftrightarrow \quad \rho(\iota, \otimes(x,y)) \in F \\
&\Longleftrightarrow \quad \rho(J_y(x), \sigma_{n+1} \cdots \sigma_k) \in F \\
&\Longleftrightarrow \quad J_y(x) \in Q_y \\
&\Longleftrightarrow \quad J_{y'}(x) \in Q_{y'} \\
&\Longleftrightarrow \quad \rho(J_{y'}(x), \delta_{n+1} \cdots \delta_l) \in Q_{y'} \\
&\Longleftrightarrow \quad \mathcal{A} \models \phi(x,y').
\end{aligned}$$

Then $c_{n,y} = c_{n,y'}$ as claimed. Thus the number of functions of the form $c_{n,y}$ ($y \in A \cap \Sigma^{>n}$) is at most the number of distinct pairs of the type (J_y, Q_y). Now the number of Q_y is at most $2^{|Q|}$, so we concentrate on bounding the number of distinct functions J_y by $O(|A^{\leq n}|)$.

Note that J_y depends only on the first n letters of y in the sense that for $|v| \geq n$ and $w, w' \in \Sigma^\star$, $J_{vw} = J_{vw'}$. Now every $y \in A \cap \Sigma^{>n}$ can be decomposed as $y = vw$ with $|v| = n$. But the first part of Lemma 3.10 says that there are $O(|A^{\leq n}|)$ many such y, and so the result follows. $\Box$

We give several applications of this theorem. First we apply the theorem to random graphs.

Corollary 4.14 (independently [7]) *The random graph has no automatic presentation.*

Proof. Let (A, E) be an automatic presentation of the random graph and let $\Phi(x,y)$ be $E(x,y)$. For every partition X_1, X_2 of the set $A^{\leq n}$ of vertices there exists a vertex y such that for all $x_1 \in X_1$ and $x_2 \in X_2$ it holds that $(x_1, y) \in E$ and $(x_2, y) \notin E$. Hence, for every n, we have $2^{|A^{\leq n}|}$ number of functions of type $c_{n,y}$, contradicting Theorem 4.13. Hence the random graph has no automatic presentation. $\Box$

Corollary 4.15 *Let $\mathcal{A}$ be a random structure of a signature L, where L contains at least one non-unary symbol. Then $\mathcal{A}$ does not have an automatic presentation.*

Proof. Let R be a non-unary predicate of L of arity k. Consider the following formula $E(x, y) = R(x, y, a_1, \ldots, a_{k-2})$, where $a_1, \ldots, a_{k-2}$ are fixed constants from the domain of $\mathcal{A}$. It is not hard to prove that (A, E) is isomorphic to the random graph. But if $\mathcal{A}$ is automatically presentable then so is the random graph (A, E), hence contradicting the previous corollary. $\Box$

The next goal is to show that the random K_p-free graph has no automatic presentation. For this one needs to have a finer analysis than the one for the random graph. Let $\mathcal{F} = (V, E)$ be a finite graph. For a vertex v, write $E(v)$ for the set of vertices adjacent to v. The *degree* of a vertex is the cardinality of $E(v)$. Write $\Delta(\mathcal{F})$ for the maximum degree over all the vertices of F. Call a subgraph $\mathcal{G}$ with no edges an *independent* graph. Let $\alpha(\mathcal{F})$ be the number of vertices of a largest independent subgraph of $\mathcal{F}$.

K_p denotes the complete graph on p vertices; that is, there is an edge between every pair of vertices. A graph is called *K_p-free* if it has no subgraph isomorphic to K_p.

Lemma 4.16 *For every $p \geq 3$, there is a polynomial $Q_p(x)$ of degree $p-1$ so that if $\mathcal{F}$ is a finite K_p-free graph then $Q_p(\alpha(\mathcal{F})) \geq |F|$.*

Proof. We first prove that for every finite graph $\mathcal{F}$,

$$\alpha(\mathcal{F}) \geq |F|/(\Delta(\mathcal{F}) + 1). \tag{1}$$

Let $\mathcal{G}$ be an independent subgraph of $\mathcal{F}$ with a maximal number of vertices. That is, $\alpha(\mathcal{F}) = |\mathcal{G}|$. For every $d \in G$ let $N(d) = E(d) \cup \{d\}$, where $E(d)$ is the set of vertices in $\mathcal{F}$ adjacent to d. Then since $\mathcal{G}$ is maximal, for every $x \in F$ there is some (not necessarily unique) $d \in G$ such that $x \in N(d)$. Hence $\mathcal{F} = \cup_{d \in G} N(d)$. But $|N(d)|$ equals the degree (in $\mathcal{F}$) of d plus one, and so the largest cardinality amongst the $N(d)$'s is at most $\Delta(\mathcal{F}) + 1$. Hence $|F| \leq |G| \times (\Delta(\mathcal{F}) + 1)$ as required.

The lemma is proved by induction on p. We will show that $Q_p(x) = \Sigma_{i=1}^{p-1} x^i$. For the case $p = 3$ note that for every vertex v, the subgraph on domain $E(v)$ is independent. For otherwise if $x, y \in E(v)$ were joined by an edge then the subgraph of $\mathcal{F}$ on $\{x, y, v\}$ is K_3. In particular, $\alpha(\mathcal{F}) \geq \Delta(\mathcal{F})$. Combining this with Inequality (1), we get $\alpha(\mathcal{F})[\alpha(\mathcal{F}) + 1] \geq |F|$ as required.

For the inductive step, let $\mathcal{F}$ be a K_p-free graph with $p > 3$. For every vertex v, the set $E(v)$ is K_{p-1}-free for otherwise the subgraph of $\mathcal{F}$ on $E(v) \cup \{v\}$ has a copy of K_p. Applying the induction hypothesis to $E(v)$ such that $|E(v)| = \Delta(\mathcal{F})$, we get that $E(v)$ must have an independent set X so that $Q_{p-1}(|X|) \geq |E(v)|$. But X is also independent in $\mathcal{F}$ so $Q_{p-1}(\alpha(\mathcal{F})) \geq \Delta(\mathcal{F})$. Combining this with Inequality 1, we get that $\alpha(\mathcal{F})[Q_{p-1}(\alpha(\mathcal{F})) + 1] \geq |F|$. Hence $Q_p(\alpha(\mathcal{F})) \geq |F|$ as required. $\Box$

Corollary 4.17 *For $p \geq 3$, the random K_p-free graph is not automatically presentable.*

Proof. Fix $p \geq 3$ and let (D, E) be a copy of the random K_p-free graph. Then for every K_{p-1}-free subset $K \subset D^{\leq n}$ there exists an $x \in D$ that is connected to every vertex in K and none in $D^{\leq n} \setminus K$. So let $\mathcal{G}_n$ be an independent subgraph of $\mathcal{D}^{\leq n}$ so that $Q_p(|G_n|) \geq |D^{\leq n}|$ as in the lemma. So letting $\Phi(x, y)$ be $E(x, y)$, for a fixed n the number of functions $c_{n,y}$ as y varies over D is at least $2^{|G_n|}$ which is not linear in $|D^{\leq n}|$. Hence by Theorem 4.13 the random K_p graph $\mathcal{D}$ is not automatically presentable. $\Box$

As the fourth application we prove that the random partial order $\mathcal{U}$ does not have an automatic presentation. For the proof we need the following combinatorial result that connects the size of a finite partial order $(B, \leq)$ with the cardinalities of its chains and anti-chains.

Lemma 4.18 (Dilworth) *Let $(B, \leq)$ be a finite partial order of cardinality n. Let a be the size of largest anti-chain in $(B, \leq)$ and let c be the size of the largest chain in $(B, \leq)$. Then $n \leq ac$.*

Proof. For $1 \leq i \leq c$ define X_i as the set of all elements x such that the size of the largest chain in the subpartial order $(\uparrow x) = \{y \in B \mid x \preceq y\}$ is i. Then the X_i's partition B. Moreover if $a \prec b$ and $b \in X_i$ then the size of the largest chain in $(\uparrow a)$ is $> i$. Hence each X_i is an anti-chain. Thus $\mathcal{B}$ can be partitioned into exactly c many anti-chains. If a is the size of the largest anti-chain in $\mathcal{B}$ then $n \leq ac$ as required. $\Box$

Corollary 4.19 *The random partial order $\mathcal{U} = (U, \leq)$ has no automatic presentation.*

Proof. Recall that the random partial order has the following property

1. If Z is a *finite* anti-chain of $\mathcal{U}$, and X and Y partition Z, then there exists an element $z \in U$ such that $z > x$ for every $x \in X$, and z is not comparable with y for every $y \in Y$.
2. If Z is a *finite* chain of $\mathcal{U}$ with least element x and largest element y, then there exists an element $z \in U$ such that $z > x$ and $z < y$ and z is not comparable with every $v \in X \setminus \{x, y\}$.

Assume that $\mathcal{U}$ has an automatic presentation $(A, \leq)$. The formula $\Phi(x, y)$ is $x \leq y \vee y \leq x$. Now let us take $A^{\leq n}$.

Let Z be an anti-chain of $A^{\leq n}$. Consider a subset X of Z. There exists an element $y \in U$ such that for every $x \in X$, $y > x$ and for every $x' \notin Z \setminus X$, element y is not comparable with x'. From this we conclude that

$$(\star) \qquad \#(\text{functions of type } c_{n,y}) \geq 2^{|Z|}.$$

Let Z be a chain of $A^{\leq n}$ with least element v and largest element w. Then there exists an element $y \in U$ such that $y > v$ and $y < w$ and y is not comparable with x for every $x \in Z$. From this we conclude that

$$(\star\star) \qquad \#(\text{functions of type } c_{n,y}) \geq \binom{|Z|}{2}.$$

Let X be the largest anti-chain and Y be the largest chain of $A^{\leq n}$ with cardinalities a and c, respectively. Using Dilworth Lemma, substituting a for $|Z|$ in ($\star$), and c for $|Z|$ in ($\star\star$), one can, with a little algebra, derive a contradiction to the bound in the statement of Theorem 4.13. $\square$

5 The Isomorphism Problem

The results in the previous sections give us hope that one can characterise automatic structures for certain classes of structures, e.g. Boolean algebras. However, in this section we prove that the isomorphism problem is Σ^1_1-complete, thus showing that the problem is as hard as possible when considering the class of *all* automatic structures. Then **complexity of the isomorphism problem** for automatic structures consists in establishing the complexity of the set $\{(\mathcal{A}, \mathcal{B}) \mid \mathcal{A}$ and $\mathcal{B}$ are automatic structures and $\mathcal{A}$ is isomorphic to $\mathcal{B}\}$.

Let $\mathcal{M}$ be a Turing machine over input alphabet Σ. Its configuration graph $C(\mathcal{M})$ is the set of all configurations of $\mathcal{M}$, with an edge from c to d if T can move from c to d in a single transition. The following is an easy lemma:

Lemma 5.1 *For every Turing machine T the configuration graph $\mathcal{C}(T)$ is automatic. Further, the set of all vertices with outdegree (indegree) 0 is FA-recognisable.*

Definition 5.2 *A Turing machine $\mathcal{R}$ is* **reversible** *if every vertex in $C(\mathcal{R})$ has both indegree and outdegree at most one.*

Now let $\mathcal{R}$ be a reversible Turing machine. Consider its configuration space $C(\mathcal{R})$. The machine $\mathcal{R}$ can be modified so that it only halts in an accepting state; so, instead of halting in a rejecting state, it loops forever. Let x be a configuration of $\mathcal{R}$. Consider the sequence: $x = x_0,\ x_1,\ x_2,\ \ldots$ such that $(x_i, x_{i+1}) \in E$, where E is the edge relation of the configuration space. Call this sequence the **chain defined by** x. We say that x is the **base** of chain X. If x does not have a predecessor then the chain defined by x is maximal. Since $\mathcal{R}$ is reversible, the configuration space $C(\mathcal{R})$ is a disjoint union of maximal chains such that each chain is either finite, or isomorphic to $(\mathbb{N}, S)$, or isomorphic to $(\mathbb{Z}, S)$, where $(x, y) \in S$ iff $y = x + 1$. It is known that every Turing machine can be converted into an equivalent reversible Turing machine (see for example [3]). Our next lemma states this fact and provides some additional structural information about the configuration space of reversible Turing machines:

Lemma 5.3 *A deterministic Turing machine can be converted into an equivalent reversible Turing machine $\mathcal{R}$ such that every maximal chain in $C(\mathcal{R})$ is either finite or isomorphic to $(\mathbb{N}, S)$.*

Denote by $\mathbb{N}^\star$ the set of all finite strings from $\mathbb{N}$. A set $T \subset \mathbb{N}^\star$ is a *special tree* if T is closed downward, namely $xy \in T$ implies $x \in T$, for $x, y \in \mathbb{N}^\star$. We view these special trees as structures of the signature E, where $E(x, y)$ if and only if $y = xz$ for some $z \in \mathbb{N}$. Thus, for every $x \in \mathbb{N}^\star$ the set $\{y \mid E(x, y)\}$ can be thought as the set of all *immediate successors of* x.

A special tree T is *recursively enumerable* if the set T is the domain of the function computed by a Turing machine. We will use the following fact from computable model theory [9, Thm 3.2].

Lemma 5.4 *The isomorphism problem for recursively enumerable special trees is Σ_1^1-complete. In fact, the trees can be chosen to be subtrees of $\{2n : n \in \mathbb{N}\}^*$.*

It is clear from the proof in [9] that the trees obtained are special (namely, subtrees of $\mathbb{N}^\star$). By a mere change of notation one obtains subtrees of $\{2n : n \in \mathbb{N}\}^*$.

Lemma 5.5 *The special tree $\mathbb{N}^\star$ has an automatic presentation $\mathcal{A}_1$.*

Proof. Consider the prefix relation $\leq_p$ on the set of all binary strings. The tree $\mathbb{N}^\star$ is isomorphic to the automatic successor tree $\mathcal{A}_1 = (\{0,1\}^\star 1 \cup \{\lambda\}; E_1)$, where E_1 is the set of all pairs (x, y) such that y is the immediate $<_p$-successor of x. The isomorphism is established via the computable mapping sending $n_1 \ldots n_k$ to $0^{n_1}1 \ldots 0^{n_k}1$ and the root to λ. $\square$

Define $\mathcal{S} = \{0,1\}^\star 1 \cup \{\lambda\}$. From now on we make the following conventions:

1. All special trees we consider will be viewed as subsets of $(\mathcal{S}, \leq_p)$.
2. The set of inputs for all Turing machines considered are strings from $\mathcal{S}$. Each $w \in \mathcal{S}$ is identified with the initial configuration starting from w.
3. All Turing machines considered are reversible.
4. The domains of the functions computed by the Turing machines are assumed to be downward closed. Thus, these domains form recursively enumerable special trees.

Our goal is to transform every recursively enumerable tree $T \subset \mathcal{S}$ into an automatic structure $\mathcal{A}_\mathcal{R}$, where $\mathcal{R}$ is the reversible machine with domain T. The idea is to attach computations of $\mathcal{R}$ to the initial configurations in $\mathcal{S}$. For all recursively enumerable trees T_1 and T_2, T_1 and T_2 are isomorphic if and only if the automatic structures $\mathcal{A}_{\mathcal{R}_1}$ and $\mathcal{A}_{\mathcal{R}_2}$ are isomorphic. To ensure this we also have to attach infinitely many chains of each finite length to the initial configurations, for in that case the length of a halting computation does not make a difference.

Let $\mathcal{R}$ be a reversible Turing machine. A **chain** is an initial segment of $(\mathbb{N}, S)$. Let $\mathcal{J}$ be the graph consisting of infinitely many finite chains of every finite length. We denote by J the set of vertices of $\mathcal{J}$, and the bases of the chains in $\mathcal{J}$ by $b_1, b_2, \ldots$. The following is not hard to prove:

Lemma 5.6 *The graph $\mathcal{J}$ has an automatic presentation.*

To construct $\mathcal{A}_{\mathcal{R}}$, with every $w \in A_1$ associate a graph $\mathcal{J}_w$ defined as follows. The vertex set of $\mathcal{J}_w$ consists of all $\{(w,j) \mid j \in J\}$ and edges between (w,j) and (w,j') if and only if (j,j') is an edge in $\mathcal{J}$. Put connecting edges from w into each (w,b_i). Let $\mathcal{A}_2$ be the graph consisting of $\mathcal{A}_1$ and every $\mathcal{J}_w$ and the connecting edges. Clearly, the graph $\mathcal{A}_2$ is automatic.

To the set A_2 add all the configurations of the Turing machine $\mathcal{R}$ and all the edges of the configuration space of $\mathcal{R}$. Note that each $w \in \mathcal{S}$, which is an initial configuration of $\mathcal{R}$ is by Convention 2 already in A_2. In addition, add a disjoint automatic copy of the graph $\mathcal{J}$, and an automatic copy of infinitely many infinite chains. Call all chains added at this stage **junk chains**. The resulting graph is denoted by $\mathcal{A}_{\mathcal{R}}$.

The proof of the following is straightforward and is omitted.

Lemma 5.7 *The graph $\mathcal{A}_{\mathcal{R}}$ is automatic.*

We summarise the structure of the graph $\mathcal{A}_{\mathcal{R}}$. Firstly, in $\mathcal{A}_{\mathcal{R}}$ there are infinitely many junk chains of every (finite and infinite) length. An **isolated chain** of $\mathcal{A}_{\mathcal{R}}$ is one in which every vertex, except the base, has exactly one successor in $\mathcal{A}_{\mathcal{R}}$.

With each element w in $\mathcal{A}_1$ there is an associated structure $\mathcal{J}_w$, which consists of infinitely many isolated chains of every finite length, and no infinite isolated chain. Finally with every initial configuration w (which is an element of $\mathcal{A}_{\mathcal{R}}$ and belongs to the infinitely branching tree $\mathcal{A}_1$) there is an isolated chain with base w that is the computation path from the configuration space of Turing machine T. These observations prove the following.

Lemma 5.8 *Let w be an initial configuration of $\mathcal{A}_{\mathcal{R}}$. Then w is the base of infinitely many isolated chains of every finite length. Also the Turing machine T halts on w if and only if there is no infinite isolated chain with base w. In case T does not halt on w there is exactly one infinite isolated chain with base w.*

Lemma 5.9 *Let T_1 and T_2 be computable trees which are domains of reversible Turing machines $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively. Then the trees T_1 and T_2 are isomorphic if and only if the automatic graphs $\mathcal{A}_{\mathcal{R}_1}$ and $\mathcal{A}_{\mathcal{R}_2}$ are isomorphic.*

Proof. First note that the domain T of $\mathcal{R}$ consists of all w in $\mathcal{A}_{\mathcal{R}}$ that are initial configurations and are not the base of an infinite isolated chain. Hence T may be thought of as a subgraph of $\mathcal{A}_{\mathcal{R}}$. Now suppose $\mathcal{A}_{\mathcal{R}_1}$ is isomorphic to $\mathcal{A}_{\mathcal{R}_2}$ via the map ϕ. Note that w is the base of an infinite isolated chain if and only if $\phi(w)$ is the base of an infinite isolated chain. Hence ϕ is an isomorphism between the trees T_1 and T_2 viewed as subgraphs of $\mathcal{A}_{\mathcal{R}_1}$ and $\mathcal{A}_{\mathcal{R}_2}$ respectively.

Conversely assume T_1 and T_2 are isomorphic via ψ (as before we may assume that T_i is a subgraph of $\mathcal{A}_{\mathcal{R}_i}$). Consider the sets I_1 and I_2 of all junk chains in $\mathcal{A}_{\mathcal{R}_1}$ and $\mathcal{A}_{\mathcal{R}_2}$. There is an isomorphism between I_1 and I_2, since both consist of infinitely many chains of every length (finite and infinite), independently of the chains of the configuration graphs that are *not* defined by initial configurations.

Therefore it suffices to establish an isomorphism extending ψ on the rest of the structure.

If $w \in T_1$ then there is an isomorphism between all the isolated chains with base w and all the isolated chains with base $\psi(w)$. Indeed each is the base of infinitely many isolated chains of every finite length and no isolated chain of infinite length, independently of the (possibly different) lengths of the finite computations of R_1 on w and R_2 on $\psi(w)$. Hence extend ψ from the chains defined by elements of T_1 to the chains defined by elements of T_2.

Suppose $w \in T_1$ and consider the set S_1 of immediate successors of w that are initial configurations but not in T_1. Similarly write S_2 for the set of immediate successors of $\psi(w)$ that are initial configurations but not in T_2. By the extra condition in Lemma 5.4 that the trees be subtrees of $\{2n : n \in \mathbb{N}\}^*$, both S_1 and S_2 are infinite. So we may extend ψ to those immediate successors by adding a bijection between S_1 and S_2.

Finally, for any initial configurations $v_1 \in \mathcal{S} - T_1$ and $v_2 \in \mathcal{S} - T_2$, there is an isomorphism between the nodes in $\mathcal{S}$ extending v_1 and the ones extending v_2. This isomorphism extends to the attached chains (as there is one infinite isolated chain and infinitely many isolated chains of each finite length). So we may extend ψ to an isomorphism of $\mathcal{A}_{\mathcal{R}_1}$ and $\mathcal{A}_{\mathcal{R}_2}$. □

Hence we have reduced the isomorphism problem for recursively enumerable trees to the isomorphism problem for automatic graphs. The main result now follows.

Theorem 5.10 *The isomorphism problem for automatic structures is Σ_1^1-complete.*

References

[1] M. Benedikt and L. Libkin. Tree extension algebras: logics, automata, and query languages. In *Proceedings of the 17th Annual IEEE Symposium on Logic in Computer Science* (LICS), pages 203–212, 2002.

[2] M. Benedikt, L. Libkin, T. Schwentick and L. Segoufin. A Model-Theoretic approach to regular string relations. In *Proceedings of the 16th Annual IEEE Symposium on Logic in Computer Science* (LICS), pages 431–440, 2001.

[3] C. H. Bennett. Logical reversibility of computation. *IBM Journal of Research and Development*, 17(6):525–532, November 1973.

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

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

[6] J. W. Cannon, D. B. H. Epstein, D. F. Holt, S. V. F. Levy, M. S. Paterson and W. P. Thurston. *Word processing in groups*. Jones and Bartlett, 1992.

[7] C. Delhommé. Automaticité des ordinaux et des graphes homogènes. *Comptes Rendus Mathematique*, 339(1):5–10, 2004.

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

[9] S. S. Goncharov and J. F. Knight. Computable structure and non-structure theorems. *Algebra and Logic*, 41(6):351–373, 2002.

[10] W. A. Hodges. *Model Theory*. Cambridge University Press, 1993.

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

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

[13] B. Khoussainov, S. Rubin and F. Stephan. Automatic linear orders and trees. *ACM Transactions on Computational Logic (TOCL)*, 6(4), 2005.

[14] L. Libkin and F. Neven. Logical definability and query languages over unranked trees. *Proceedings of the 18th Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 178–187, 2003.

[15] A. Nies. Describing Groups. To appear, available at `http://www.cs.auckland.ac.nz/~nies/papers/`.

[16] S. Rubin. *Automatic Structures*. Phd thesis, University of Auckland, 2004.