Automatic Presentations of Structures

Bakhadyr Khoussainov and Anil Nerode

Introduction

In this paper we introduce the systematic study of presentations of algebraic structures by finite automata. We call these automatic structures.

The study of recursive presentations of algebraic structures was initiated by Froehlich and Shepherdson [8], Rabin [14], and Mal'cev [11]. Since then, recursive algebra has been an active area of study by researchers associated with Nerode in the U. S. and Ershov in Russia. A **recursive structure** is a countable structure equipped with Turing machines for deciding equality and the other atomic relations. The ordering of rationals, vector spaces over rational numbers, absolutely free algebras, free groups, and finitely presented algebras with decidable word problems have obvious recursive presentations. Recursive Boolean algebras, linear orderings, abelian groups, vector spaces, fields, lattices, and other structures have been studied extensively.

In the late 1980's Nerode, Remmel, and Cenzer [15] [16] developed a corresponding theory of p-time structures. These are recursive structures which are presented by polynomial time recursive functions. This has also become an active area of research. For example, Remmel proved that any recursive, purely relational, structure is recursively isomorphic to a p-time structure, and, as a corollary, that any recursive Boolean algebra has a polynomial time presentation.

In this paper we further restrict the recursive functions of the presentaions and insist they be given by finite automata. We obtain a very fine grain theory of automatic, or **automaton presentable** structures. That is, these are structures provided with finite automata for deciding equality on the domain and the atomic relations of the structure.

We note that automatic groups in a closely related sense are an active object of study ([6]) growing from the need to have feasible calculations in 3-manifold theory. Epstein, Cannon, Thurston have developed the theory of automatic groups motivated by performing computations on groups associated with 3-manifolds. They consider finitely generated groups with generators $g_1, \ldots, g_n$. Each generator g_i naturally defines a unary operation f_i on the domain of G by the right multiplication, that is $f_i(x) = xg_i$, where $x \in G$. Thus, with the finitely generated group G one can associate the unary structure $(G, f_1, \ldots, f_n)$. They call the group G automatic ([6]) if the corresponding unary structure $(G, f_1, \ldots, f_n)$ is automatic in our sense below. They do not impose the requirement that the binary group operation be described by a finite automaton.

They prove, for instance, that any automatic group is finitely presented [6]. Automatic groups in the sense of Epstein, Cannon, and Thurston, regarded as unary structures as above, are automatic in our sense.

Research on automatic structures, unlike research on recursive and p-time structures, concentrates on positive results. The results of Epstein,Cannon, Thurston show the usefulness of automaton presentations. If one has an automaton presentation of a structure $\mathcal{A}$, one can perform automaton computations on the structure. As an example, let P_S be the following problem: is there a recursive procedure which, when applied to a first order definition of a relation S on $\mathcal{A}$, yields an algorithm for deciding P. To solve this problem, we want to find presentations of $\mathcal{A}$ in which computations of atomic relations are governed by finite automata. In such a presentation, these computations can be performed in real time. If we find such a presentation, we can transform the problem P_S into a problem about finite automata. Since finite automata possess many decidability properties, we can deduce the decidability of many problems such as P_S. This, in turn, leads to consideration of the complexity of problems about automatic structures.

Here are several basic questions.

- What is an automatic presentation?
- Which structures are automaton presentable?
- What does automaton presentablity say about the structure?
- What are the easiest standard structures that have or lack automaton presentations?
- What is the complexity of problems formulated over automatic structures?

We define the notions of strongly automatic, automatic, and asynchronously automatic presentations. We provide many examples of automatic presentations of linear orderings, vector spaces, abelian groups, permutation structures, etc. Though these examples are not startling, they show that automatic presentations play a basic role underlying many recursive and polynomial time structures. We obtain an algebraic characterization of automatic structures, by introducing many sorted finite automata. This leads to a natural generalization of the Myhill-Nerode theorem charcterizing finite automaton recognizable languages by congruences of finite index. We also characterize structures which possess strongly automatic presentations. At the end we discuss automatic isomorphism types.

1 Preliminary Definitions

Consider structures of the form $(A, f_0^{n_0}, \ldots, f_k^{n_k}, P_0^{m_0}, \ldots, P_s^{m_s}, c_0, \ldots, c_t)$, where A is the **domain**, each $f_i^{n_i}$ is an **operation** of arity n_i on A, each $P_j^{m_j}$ is a **predicate** of arity m_j on A, and each **constant** c_i belongs to A. We suppose that the domain A is at most a countable set. The sequence

$$(f_0^{n_0}, \ldots, f_k^{n_k}, P_0^{m_0}, \ldots, P_s^{m_s}, c_0, \ldots, c_t)$$

is called the **signature** of the structure $\mathcal{A}$. Given a structure $\mathcal{A}$ of signature, form a new structure

$$\mathcal{A}_R = (A, F_0^{n_0+1}, \ldots, F_k^{n_k+1}, P_0^{m_0}, \ldots, P_s^{m_s}, c_0, \ldots, c_t),$$

where for all $a_1, \ldots, a_{n_i+1} \in A$, $F_i^{n_i+1}(a_1, \ldots, a_{n_i+1})$ iff $f_i^{n_i+1}(a_1, \ldots a_{n_i}) = a_{n_i+1}$. This makes the structure $\mathcal{A}_R$ **relational**. We often identify the structure $\mathcal{A}$ with $\mathcal{A}_R$.

Let Σ be a finite alphabet. Form the set $\Sigma^\star$ of all finite words of the alphabet Σ. A **(nondeterministic) finite automaton** over the alphabet Σ is a quadruple $\Omega = (S, I, \Delta, F)$, where S is the finite non-empty **set of states**, I is a non-empty subset of S, called the set of **initial states**, $\Delta \subseteq S \times \Sigma \times S$ is a non-empty set, called **the transition table**, F is a subset of S, called the set of **final states**. Thus, Δ can be viewed as a mapping from $S \times \Sigma$ to $P(S) = \{S' \mid S' \subset S\}$. The mapping Δ can be extended naturally to a mapping from $S \times \Sigma^\star$ to $P(S)$. If there is no confusion, denote this extension by Δ too. For convenience we present a finite state automaton Ω as a directed graph. The nodes of the graph are the states of the automaton Ω. The edge relation E on the nodes is defined as follows. There exists an edge connecting node s with node s' if and only if $(s, \sigma, s') \in \Delta$ for some $\sigma \in \Sigma$. We label this edge σ. Thus, $E = \{(s, s') \mid \exists\sigma \in \Sigma((s, \sigma, s') \in \Delta)\}$. The automaton Ω **accepts** $\sigma_1 \ldots \sigma_n$ if and only if there exists a path $s_0 \ldots s_n$ in the graph presenting the automaton such that $s_0 \in S_0$, $s_n \in F$, and for each i $(s_i, \sigma_i, s_{i+1}) \in \Delta$. We call such a path a **computation** of Ω on input $\sigma_1 \ldots \sigma_n$. The **behavior** of the automaton Ω is defined as the set $L(\Omega)$ of all words accepted by Ω. A set, or equivalently language, $D \subseteq \Sigma^\star$ is **finite automaton (FA) recognizable**, or simply **recognizable**, if there exists a finite automaton Ω such that $D = L(\Omega)$. It is well known that the set of all finite automata recognizable sets forms a Boolean algebra and that the emptiness problem for finite automata is decidable.

Dealing with an automatic structure $\mathcal{A}$, it is necessary to have an automaton that recognizes the domain of the structure. To do this we have to **present** elements of the structure as words of a finite alphabet. Thus, suppose that a structure $\mathcal{A}$ is given. An **automatic presentation of the domain** of $\mathcal{A}$ is a surjective mapping $\nu : D \to A$, where D is a recognizable subset of $\Sigma^\star$. If $a \in A$ and $\nu(\alpha) = a$, we say that α **presents** the element a. Having the mapping ν, we formulate three problems.

1. Find a procedure which, for any two words α, $\beta \in D$ decides whether $\nu(\alpha) = \nu(\beta)$.

2. Find a procedure which, for each predicate $P_j^{m_j}$, and all $\alpha_1, \ldots, \alpha_{m_j} \in D$, decides whether $P_j^{m_j}(\nu(\alpha_1), \ldots, \nu(\alpha_{m_j}))$ holds.

3. Find a procedure which for each operation $f_i^{n_i}$, and all $\alpha_1, \ldots, \alpha_{n_i+1} \in D$ decides whether $F_i^{n_i+1}(\nu(\alpha_1), \ldots, \nu(\alpha_{n_i+1}))$ holds.

Informally, a presentation ν is an **automatic presentation** of $\mathcal{A}$ if there exist automata for deciding the above three problems. This considerations suggest the definition of automata recognizable relations over $\Sigma^\star$.

1. n-variable Strong Automata.

The intuitive meaning of the concept of n-variable strong automaton is as follows. The inputs for such an automaton are n-tuples of words over Σ. Given an input, the automaton acts on each component of the input exactly as a finite automaton. Computations at different components of the input are independent.

Definition 1.1 *A* **strong n-variable automaton** Ω_n *on* Σ *is a system* (S, S_0, Δ, F), *where* S, S_0, Δ *are as for finite automata and* F *is a subset of* S^n, *called the set of* **final states**.

Let $(\alpha_1, \ldots, \alpha_n) \in (\Sigma^\star)^n$ and let α_i be $\sigma_{i1} \ldots \sigma_{im_i}$, $i = 1, \ldots, n$. A sequence $(s_{11}, \ldots, s_{n1}), (s_{12}, \ldots, s_{n2}), \ldots, (s_{1m_1}, \ldots, s_{nm_n})$ is a **computation** of the automaton Ω_n on $(\alpha_1, \ldots, \alpha_n)$ if $(s_{11}, \ldots, s_{n1}) \in S_0^n$, and for each i, the sequence $s_{i1}, \ldots, s_{im_i}$ is a computation of Ω_n on the component α_i according to the transition table. The n-tuple $(\alpha_1, \ldots, \alpha_n)$ is **accepted** by the n-variable strong automataion Ω_n if there exists a computation

$$(s_{11}, \ldots, s_{n1}), (s_{12}, \ldots, s_{n2}), \ldots, (s_{1m_1}, \ldots, s_{nm_n})$$

of the automaton on the n–tuple such that $(s_{1m_1}, \ldots, s_{nm_n}) \in F$. A relation L of airity n on $\Sigma^\star$ is **strongly recognizable** if there exists an n-variable strong automaton Ω_n on Σ such that the set of all n-tuples accepted by this automaton is exactly L.

2. n-variable Automata. Let Σ be a finite alphabet. Suppose that the symbol $\diamond$ does not belong to Σ. Take words $\alpha_i = \sigma_{i1} \ldots, \sigma_{in_i}$ of the alphabet Σ, where $i = 0, \ldots, n-1$. The **convolution** $\alpha_0 \star \ldots \star \alpha_{n-1}$ of these words is defined in the following way. If for all $i, j < n$ $n_i = n_j$, then the convolution is

$$(\sigma_{01}, \ldots, \sigma_{n-11}), \ldots, (\sigma_{0n_0}, \ldots, \sigma_{n-1n_0}).$$

Otherwise, let m be the maximal length of the words $\alpha_0, \ldots, \alpha_{n-1}$. Add to the right end of each α_i the necessary number of symbols $\diamond$ to get words of the length m. Call these new words α_i', $i = 0, \ldots, n-1$. The **convolution** of these n-tuples is $\alpha_0' \star \ldots \star \alpha_{n-1}'$. This convolution is a word of the alphabet $(\Sigma \cup \{\diamond\})^n$. Thus, for any n–ary relation R on $\Sigma^\star$ we can consider the subset $R^\star \subset (\Sigma \cup \{\diamond\})^n$ obtained from R using convolution, that is,

$$R^\star = \{\alpha_0 \star \ldots \star \alpha_{n-1} \mid (\alpha_0, \ldots, \alpha_{n-1}) \in R\}.$$

Definition 1.2 1) **An n-variable automaton** *on* Σ *is a finite automaton over the alphabet* $(\Sigma \cup \{\diamond\})^n$. 2) *An n-ary relation R in $\Sigma^\star$ is* **n-recognizable,** *if $R^\star$ is recognizable by an n–variable automaton.*

3. Asynchronous Automata. Another recognizability notion for relations on $\Sigma^\star$ based on the notion of an asynchronous automaton. Let $P(Q)'$ be the set of all non-empty subsets of Q, let $P(i)'$ be short for $P(\{1, \ldots, i\})'$, $\diamond \notin \Sigma$, and let $\Sigma' = \Sigma \cup \{\diamond\}$.

Definition 1.3 **An** n**-variable asynchronous atomation** Ω *on* Σ *is a quadruplet* (S, S_0, Δ, F)*, where* S *is* **the set of states**, S_0 *is the set of* **initial states**, $F \subseteq S$ *is* **the set of final states**, *and* $\Delta : S \times (\Sigma')^n \rightarrow P(S) \times P(P(n)'))$ *is a partial mapping called the* **transition table** *such that:*

1. *For all* $\sigma \in (\Sigma')^n$, $s \in S$ *if* $\sigma = (\sigma_1, \ldots, \sigma_n)$, $\sigma_i = \diamond$ *for some* i, $\Delta(s, \sigma) = (L, R)$, *and* $J \in R$, *then* $i \notin J$.

2. *For all* $s \in S$, $\Delta(s, \sigma)$ *is undefined if and only if* $\sigma = (\diamond, \diamond, \ldots, \diamond)$.

Let us take words $\alpha_i = \sigma_{i1}, \ldots, \sigma_{in_i}$, $i = 1, \ldots, n$. The intended behaviour of an asynchronous automaton Ω on the n-tuple $\alpha = (\alpha_1, \ldots, \alpha_n)$ is as follows. The automaton begins its computation from an initial state. Suppose that that automaton is in a state $s \in S$ and that the input is $\sigma' = (\sigma_1, \ldots, \sigma_n) \in (\Sigma')^n$. Consider the pair (L, R) defined by the transition table $\Delta(s, \sigma') = (L, R)$. Then the automaton non-deterministically chooses a state $s' \in L$, a non-empty set $\{i_1, \ldots, i_k\} \in R$, and makes moves on components $\alpha_{i_1}, \ldots, \alpha_{i_k}$. We call the pair (s, s') an **elementary move** defined by σ'. Then one can naturally define the notion of a **computation** of the automaton Ω on input α. Say that Ω **accepts** α if there exists a computation of Ω on α which begins at an initial state and ends at a final state. Thus a n-ary relation R on $\Sigma^\star$ is **asynchronously recognizable** if there exists an asynchronous automaton such that the set of all n-tuples of words accepted by this automaton is exactly R.

Presentations of Structures. Let a structure $\mathcal{A}$ of the signature be given. The following are definitions of automatic presentations. The next section gives many examples.

Definition 1.4 *Let* $\nu : D \rightarrow A$ *be a surjective mapping, where* D *is an automaton recognizable subset of* $\Sigma^\star$. *The mapping* ν *is respectively an* **automatic, (strong automatic, asynchronous automatic) presentation of** $\mathcal{A}$ *if* ν *satisfies the following conditions:*

1. *There exists a* 2*–variable automaton (*2*–variable strong automaton, asynchronous automaton) which for any two words* α, $\beta \in D$ *decides whether* $\nu(\alpha) = \nu(\beta)$.

2. *For each* $j \in \{0, \ldots, s\}$, *there exists an* m_j*–variable automaton (*m_j*–variable strong automaton, asynchronous automaton) which for all* $\alpha_1, \ldots, \alpha_{m_j} \in D$, *decides whether* $P_j^{m_j}(\nu(\alpha_1), \ldots, \nu(\alpha_{m_j}))$ *holds in the structure* $\mathcal{A}$.

3. *For each* $i \in \{0, \ldots, k\}$, *there exists an* $(n_i + 1)$*–variable automaton (*(n_i+1)*–variable strong automaton, asynchronous automaton) which, for all* $\alpha_1, \ldots, \alpha_{n_i}, a_{n_i+1} \in D$, *decides whether* $F_i^{n_i}(\nu(\alpha_1), \ldots, \nu(\alpha_{n_i+1}))$ *holds in* $\mathcal{A}$.

If ν *is an automatic (strongly automatic, asynchronous automatic) presentation of the structure* $\mathcal{A}$*, then the pair* $(\mathcal{A}, \nu)$ *is a* **(strongly, asynchronous) automatic structure** *and the structure is* $\mathcal{A}$
(strongly, asynchronous) automata presentable.

2 Some Examples

Structures with Unary Predicates. These are structures of the form $\mathcal{A} = (A, P_0, \ldots, P_m)$, where each P_i is a unary predicate.

Proposition 2.1 *Every structure with unary predicates only has an automatic presentation.*

Proof. Let $\mathcal{A} = (A, P_0, \ldots, P_m)$ be a structure with each P_i a subset of domain A. Suppose that for different $i, j \leq m$, $P_i \cap P_j = \emptyset$. Consider the alphabet $\Sigma = \{0,1\}$. On the set $\omega = \{0,1\}^\star$ we can choose pairwise disjoint recognizable sets $S_0, \ldots, S_m$ such that for each $i \leq m$, $card(P_i) = card(S_i)$, and

$$card(\{0,1\}^\star \setminus (S_0 \cup \ldots \cup S_m)) = card(A \setminus (P_0 \cup \ldots \cup P_m)).$$

Then any 1-1 function from $\nu : \{0,1\}^\star \to A$ such that $\nu(S_i) = P_i$ for each i, is an automatic presentation of $\mathcal{A}$.

Suppose that $P_0, \ldots, P_n$ are arbitrary unary predicates. Then there exist pairwise disjoint subsets $B_0, \ldots, B_k$ of A with the following property: For any $i \leq m$ there exists a Boolean combination $\Phi_i(B_0, \ldots, B_k)$ of sets $B_0, \ldots, B_k$ such that $P_i = \{x \mid x \in \Phi_i(B_0, \ldots, B_k)\}$. By the previous case, the structure $(A; B_0, \ldots, B_k)$ has a automatic presentation $(\omega; S_0, \ldots, S_k)$. Since recognizable sets are closed under Boolean operations the structure

$$(\omega, \Phi_0(S_0, \ldots, S_k), \ldots, \Phi_m(S_0, \ldots, S_k))$$

is automatic and isomorphic to $\mathcal{A}$. □

Linear Orderings. Here are several examples of automaton presentable linear orderings.

Proposition 2.2 *The rational numbers with the natural linear ordering have an automatic presentation.*

Proof. Let $\Sigma = \{0,1\}$ and let D be a set such that $\alpha 101 \in D$ if and only if $\alpha \in \Sigma^\star$ and α does not have the subword 101. It is clear that D is a recognizable subset of $\Sigma^\star$. Consider the lexicorgraphic linear ordering $\preceq_l$ on the set $\Sigma^\star$. This ordering is recognizable by a 2–variable automaton. Thus the linear ordered set $(D, \preceq_l)$ is automatic. Let $\alpha 101 \in D$. Then $\alpha 101 \preceq_l \alpha 1101$ and $\alpha 00101 \preceq_l \alpha 101$. Hence $(D, \preceq_l)$ does not have maximal and minimal elements. Similarly it can be proved that $\preceq_l$ is a dense linear ordering of the set D. It follows that $(D, \preceq_l)$ is isomorphic to the the rational numbers with the natural linear ordering.□

Proposition 2.3 1) *For any natural number $n \in \omega$, the ordinal ω^n has an automatic presentation.* 2) *The ordinal ω^ω has an asynchronous automatic presentation.*

Proof. 1. Consider the alphabet $\Sigma = \{0,1\}$. To prove the first part, define the following set $D_n = \{0^{i_1}10^{i_2}1 \ldots 0^{i_{n-1}}10^{i_n} \mid i_1, \ldots, i_n \geq 1\}$. The set D_n is recognizable. Let $\alpha, \beta \in \Sigma^\star$. Then $\alpha \leq_n \beta$ if and only if there exist $\gamma, \gamma_1, \gamma_2 \in \Sigma^\star$ such that $\alpha = \gamma 1 \gamma_1$ and $\beta = \gamma 0 \gamma_2$. One can construct a 2–variable automaton which recognizes the relation $\leq_n$. The linearly ordered set $(D, \leq_n)$ is isomorphic to ω^n.

To prove the second part, define the following set D:

$$D = \{0^{i_1} 1 0^{i_2} 1 \ldots 0^{i_{k-1}} 1 0^{i_k} \mid i_1, \ldots, i_k \geq 1, k \geq 1\}.$$

Note that D contains the set D_n as a proper subset and that $D = \bigcup_n D_n$. Let $\alpha, \beta \in \Sigma^\star$. Then $\alpha \leq \beta$ if, and only if, either $\alpha \leq_n \beta$ for some n or $\alpha \in D_t$, $\beta \in D_m$, and $t < m$. The linearly ordered set $(D, \leq)$ is isomorphic to ω^ω. It can be verified that the relation $\leq$ is recognizable by a asynchronous automaton. □

The next proposition shows that the standard operations $+$ and $\times$ over linear orderings are preserved by automatic presentations. We leave a proof of this proposition to the reader.

Proposition 2.4 *Let L_1 and L_2 be linear orderings which have (asynchronous) automatic presentations. Then the linear orderings $L_1 + L_2$ and $L_1 \times L_2$ also have (asynchronous) automatic presentations.*□

Boolean Algebras. Let $(L, \leq)$ be a linear ordering. Consider the interval Boolean algebra $\mathcal{B}_L$ generated by **intervals** $[a, b) = \{x \mid a \leq x < b\}$. Any element of this algebra is a finite union of pairwise disjoint intervals. Consider the linear ordering $n \times \omega$, where n is a natural number.

Proposition 2.5 *For every n the Boolean Algebra $\mathcal{B}_{n\times\omega}$ has an asynchronous automatic presentation.*

Proof. We prove the proposition in the case $n = 2$. The remaining case $n > 2$ is similar. We have to prove that the Boolean Algebra $\mathcal{B}_{\omega+\omega}$ has an asynchronous automatic presentation. Consider the alphabet $\{0, 1, 2, \mathbf{0}, \mathbf{1}\}$. Let

$$a = [a_1, b_1) \cup [a_2, b_2) \cup \ldots \cup [a_n, b_n)$$

be an element of the algebra such that $[a_i, b_i) \cap [a_j, b_j) = \emptyset$ for $i \neq j$, and $a_1 \leq b_1 \leq \ldots \leq a_n \leq b_n$. There are several cases.

Case 1. Suppose that $b_n < \omega$. Consider the sequence

$$\nu(a) = \epsilon_0 \ldots \epsilon_{b_n} \in \{0, 1, 2\}^\star,$$

where $\epsilon_i = 1$ if $i \in \bigcup_{j=1}^n [a_j, b_j)$, and $\epsilon_i = 0$ otherwise.

Case 2. Suppose that $a_1 > \omega$. Consider the sequence

$$\nu(a) = 02\epsilon_0 \ldots \epsilon_{b_n} \in \{0, 1, 2\}^\star,$$

where $\epsilon_i = 1$ if $\omega + i \in \bigcup_{j=1}^n [a_j, b_j)$, and $\epsilon_i = 0$ otherwise.

Case 3. Suppose that there exists an $m < n$ such that $b_m < \omega$ and $a_{m+1} > \omega$. Thus

$$a = [a_1, b_1) \cup [a_2, b_2) \cup \ldots \cup [a_m, b_m) \cup \ldots \cup [a_n, b_n).$$

Define $a_1 = [a_1, b_1) \cup \cdots \cup [a_m, b_m)$ and $a_2 = [a_{m+1}, b_{m+1}) \cup \cdots \cup [a_n, b_n)$. Consider the sequence

$$\nu(a) = \nu(a_1)02\nu(a_2).$$

Case 4. Suppose that there exists an $m < n$ such that $a_m < \omega$ and $b_{m+1} > \omega$. Thus $a = [a_1, b_1) \cup [a_2, b_2) \cup \ldots \cup [a_m, b_m) \cup \ldots \cup [a_n, b_n)$. Define

$$\begin{aligned} a_1 &= [a_1, b_1) \cup \ldots \cup [a_m, a_m + 1), \\ a_2 &= [\omega + 1, b_m) \cup [a_{m+1}, b_{m+1}) \cup \ldots \cup [a_n, b_n). \end{aligned}$$

Consider the sequence $\nu(a) = \nu(a_1)12\nu(a_2)$.

We also put $\nu(\emptyset) = \mathbf{0}$ and $\nu(L) = \mathbf{1}$. Thus we have a mapping ν mapping Boolean algebra $\mathcal{B}_{\omega+\omega}$ into $\{0, 1, 2\}^\star$. This mapping is 1-1. Let D be its range. The definition of D implies that the set D is finite automaton recognizable. The operations $\cap$ and $\cup$ in the Boolean Algebra $\mathcal{B}_{\omega+\omega}$ induce the operations $+$ and $\cdot$ in the set D. It is easy to prove that the graphs of $+$ and $\cdot$ on D are recognizable by an asynchronous automaton. $\Box$

Graphs. Here we present a general construction of automatic graphs. Suppose that $T = (q_0, Q, P_T)$ is a Turing machine over the finite alphabet $A = \{a_0, \ldots, a_n\}$, where Q is the set of states, P_T is the set of commands of T, q_0 is the initial state. Define the following set D_T:

$$D_T = \{\alpha \mid \alpha \text{ is a configuration of the Turing machine } T\}.$$

The set D_T is a finite automaton recognizable. On the set D_T consider a binary relation R_T defined by

$$R_T = \{(\alpha, \beta) \mid \text{there exists a command in } P_T \text{ transforming } \alpha \text{ to } \beta\}.$$

The set R_T is recognizable by a 2–variable automaton. Define the graph $\mathcal{G}_T = (D_T, R_T)$. We get the following proposition.

Proposition 2.6 *The graph $\mathcal{G}_T = (D_T, R_T)$ is automatic.*$\Box$

Unary Structures. We consider structures $(A, f_1, \ldots, f_n)$, where each f_i is a unary operations on A. Here are two results on automatic presentations of unary structures.

Proposition 2.7 *Any free unary structure $\mathcal{A}$ has an automatic presentation.*

Proof. We introduce the set $X = \{0, 00, 000, \ldots, 0^n, \ldots\}$. Let Σ be $\{0, f_1, \ldots, f_n\}$. Define the following set D over this alphabet:

$$D = \{0^n\alpha \mid n \geq 1 \& \alpha \in \{f_1, \ldots, f_n\}^\star\}.$$

The set D is a finite automaton recognizable. Each f_i defines a unary operation, also denoted by f_i, by letting $f_i(0^n\alpha) = 0^n\alpha f_i$. It is clear that $(D, f_1, \ldots, f_n)$ is the free unary algebra with the set of generators X. By the definition of f_i, we conclude that f_i is recognizable by a 2–variable automaton. This proves the proposition.□

A unary structure $(A, f_1, \ldots, f_n)$ is **abelian** if for all $a \in A$, $i, j \leq n$, we have $f_i f_j(a) = f_j f_i(a)$.

Proposition 2.8 *Any free abelian unary structure has an automatic presentation.*

Proof. We introduce the following alphabet $\Sigma = \{0, a_1, \ldots, a_n\}$. Define the set $D = \{0^s a_1^{n_1} \ldots a_n^{i_n} \mid s, i_1, \ldots, i_n \geq 1\}$. The set D is finite automaton recognizable. For each $i \leq n$, define a unary operation f_i on the set D as follows: $f_i(x) = 0^s a_1^{n_1} \ldots a_i^{n_i+1}$ if and only if if $x = 0^s a_1^{n_1} \ldots a_i^{n_i} \ldots a_n^{i_n}$.

It is easy to see that f_i is recognizable by a 2–variable automaton. Therefore the unary structure $(D, f_1, \ldots, f_n)$ is automatic. This structure is the free abelian unary structure on the set $\{0^s \mid s \geq 1\}$ of generators.□

A **permutation structure** is a unary structure $\mathcal{A} = (A, f)$, where f is a 1-1 function defined on the set A.

Proposition 2.9 *If the length of finite cycles of f is bounded, then the permutation structure $\mathcal{A} = (A, f)$ has an automatic presentation.*

Proof. Suppose we are given a permutation structure $\mathcal{A} = (A, f)$ which satisfies the conditions of the proposition. First, suppose that f does not have any cycles of finite length. Consider the alphabet $\{0, 1, 2\}$. Define the following set D which is finite automaton recognizable:

$$D = \{1^n 0^k \mid n, k \geq 1\} \cup \{0^m 2^t \mid m \geq 1, t \geq 0\}.$$

Define on this set unary operation f' as follows.

$$f'(x) = \begin{cases} 1^{n-1}0^k & \text{if } x = 1^n 0^k \text{ and } n \geq 1, \\ 0^m 2^{t+1} & \text{if } x = 0^m 2^t \text{ and } t \geq 0. \end{cases}$$

The function f' is recognizable by a 2–variable automaton. Thus, (D, f') is a permuation structure which has infinitely many infinite cycles. Moreover (D, f') does not have cycles of finite length. From this we conclude that any permutation structure which does not have cycles of finite length has an automatic presentation.

Next, suppose that the length each cycle of $\mathcal{A} = (A, f)$ is n. Consider the set $D \subset \{0, 1\}^\star$ defined by $D = \{0^m 1^i \mid m \geq 1, i = \{1, 2, \ldots, n-1\}\}$. The set D is a finite automaton recognizable. Define f' as follows:

$$f'(x) = \begin{cases} 0^m 1^{i+1} & \text{if } x = 0^m 1^i \text{ and } i < n-1 \\ 0^m & \text{if } x = 0^m 1^{n-1}. \end{cases}$$

The function f' is recognizable by a 2–variable automaton. Thus (D, f') is a permuation structure which is isomorphic to $\mathcal{A}$.

Now consider the general case. Let $(n_1,k_1),\ldots,(n_m,k_m)$ be the sequence of all pairs such that for each $i \leq m$, $n_i, k_i \leq \omega$, and the permutation structure $\mathcal{A} = (A, f)$ has exactly k_i cycles of length n_i. Combining the previous cases, we can conclude that $\mathcal{A}$ has an automatic presentation.□

Is the hypothesis of the previous proposition necessary? Here is an example which shows that this is not a case.

Proposition 2.10 *There exists an automaton presentable permutation structure such that the set of lengths of finite cycles of this structure is not bounded.*

Proof. Define the following function f on set $\{0,1\}^*$. If $\alpha = 1^n$, then $f(\alpha) = 0^n$. Suppose that $\alpha = \beta 01^n$, where $n \geq 1$. Then $f(\alpha) = \beta 10^n$. Suppose that $\alpha = \beta 0$. Then $f(\alpha) = \beta 1$. One can check that f is recognizable by a 2–variable automaton. Note that, for each n the function forms a cycle of length 2^n.□

Vector Spaces and Abelian Groups. First consider the simplest infinite abelian group, that is, the rank-one free abelian group of rational integers $(Z, +)$.

Lemma 2.1 *The group $(Z,+)$ has an automatic presentation.*

Proof. Each integer $n \in Z$ is a word over the alphabet $\Sigma = \{0, 1, \ldots, 9, -\}$. The standard algorithm which adds two integers gives a 3-variable automaton over Σ recognizing the relation $\{(x,y,z) \mid x + y = z\}$.□

It can be proved that the direct product of any two automata presentable groups is also automaton presentable. Since any finitely generated abelian group can be written as a direct product of a finite group and finitely many copies of $(Z,+)$, we get the following proposition.

Proposition 2.11 *Any finitely generated abelian group has an automatic presentation.* □

Remark. Any finitely generated abelian $\mathcal{A} = (A, +)$ group with the generators $g_1, \ldots, g_n$ induces a unary structure $(A, f_1, \ldots, f_n)$, where $f_i(a) = a + g_i$. In [6] it is proved that $(A, f_1, \ldots, f_n)$ has an automatic presentation. That is, any finitely generated abelian group is automatic in sense of Epstein, Cannon, and Thurston. The proposition above is clearly a stronger version of this result.

Let $\mathcal{V} = (V, \oplus)$ be a vector space over field $\mathcal{F}$. Each $f \in \mathcal{F}$ induces a unary operation, also denoted by f, on the set V by setting $f(v) = vf$. Thus, we can identify the vector space $\mathcal{V}$ with the structure $(V, \oplus, f)_{f \in F}$.

Proposition 2.12 *Any countable vector space $(V, \oplus, f)_{f\in F}$ over a finite field $\mathcal{F}$ has an automatic presentation.*

Proof. Since $\mathcal{F}$ is a finite field, we can suppose that

$$\mathcal{F} = (\{0, 1, \ldots, p-1\}, +, \cdot),$$

where p is a prime number. Consider the set $D = \{0, 1, \dots, p-1\}^\star \setminus \{e\}$, where e is the empty word. Define operation $\oplus$ on this set

$$i_1 \dots i_n \oplus j_1 \dots j_m = \begin{cases} i_1 + j_1 \dots i_n + j_n \dots j_m & \text{if } n \leq m \\ i_1 + j_1 \dots i_m + j_m \dots i_m & \text{if } m \leq n \end{cases}$$

For each $k \in \{0, 1, \dots, p-1\}$ define unary operation f_k by

$$f_k(i_1 \dots i_n) = i_1 \cdot f \dots i_k \cdot f.$$

It can be seen that the relations corresponding to the operations $\oplus$ and f_k are recognizable by 3 and 2–variable automata, respectively.□

3 A Characterization of Automatic Structures

Suppose that $(D, P_0^{m_0}, \dots, P_s^{m_s})$ is a relational structure such that $D \subset \Sigma^\star$, $P_j^{m_j} \subset (\Sigma^\star)^{m_j}$, where $j = 1, \dots, s$, and Σ is a finite alphabet. In this section we present an answer to the following question. When is this structure automatic? In order to answer to this question, we refine the Myhill-Nerode theorem which characterizes finite automaton recognizable languages. It suffices to characterize relations of arity n recognizable by n-variable automata.

Suppose that R is a relation of arity n. Let $R^\star$ be the language over alphabet $(\Sigma \cup \{\diamond\})^n$ obtained from R by convolution. By the definition of automatic structure, structure (D, R) is automatic if and only if D is finite automaton recognizable and the convolution $R^\star$ is recognizable by a finite automaton over alphabet $(\Sigma \cup \{\diamond\})^n$. Thus the Myhill-Nerode theorem can be applied to characterize when structure (D, R) is automatic. But the finite automaton recognizable languages obtained by the convolution of the relations of arity n is a **proper** subclass of all finite automata recognizable languages over $\Sigma \cup \{\diamond\})^n$. The original Myhill-Nerode theorem does not give a characterization of this class. But below, using the idea behind the Myhill-Nerode theorem combined with many-sorted algebra, we give a self-contained characterization of automaton recognizable relations on $\Sigma^\star$. We remark that there have been deep and interesting investigations to characterize this class of relations [19]. Apparently our characterization of this class of such relations is different from previous ones, possibly clearer and simpler.

Let Σ be a finite alphabet. Let $\leq$ be a pre-partial ordering (reflexive and transitive binary relation) on Σ. We say that elements σ_1, σ_2 **have the same sort** if $\sigma_1 \leq \sigma_2$ and $\sigma_2 \leq \sigma_1$. Thus, the elements of Σ are sorted. Moreover, since $\leq$ is a pre-partial ordering, Σ is a finite disjoint union

$$\Sigma_1 \cup \Sigma_2 \cup \dots \cup \Sigma_k,$$

where each Σ_i contains all elements of Σ of the same sort. This induces a partial order on

$$(\{\Sigma_i \mid i = 1, \dots, k\}, \leq).$$

If $\sigma \in \Sigma_i$, then we say that σ has sort i, and we denote it by $s(\sigma)$. If $\sigma_1 \in \Sigma_i$, $\sigma_2 \in \Sigma_j$ and $\Sigma_i \leq \Sigma_j$, we say that the sort i is **weaker** then the sort j. Introduce the following system $\Sigma_{\leq}$:

$$(\Sigma, \leq, \Sigma_1, \ldots, \Sigma_k).$$

Let Σ_1 be the smallest element with respect to $\leq$. We call such a system a **many-sorted finite alphabet.**

Let $\Sigma_{\leq}$ be a many-sorted finite alphabet. Define the set $\Sigma^{\star}_{\leq}$ of **sorted finite words** as follows. The word $\sigma_1 \ldots \sigma_n$ belongs to $\Sigma^{\star}_{\leq}$ if, and only if, for each $i \leq n-1$, the sort of σ_i is weaker than the sort of σ_{i+1}. We take it that the empty word has the weakest sort. Since the alphabet Σ is sorted we can also sort the words in $\Sigma^{\star}_{\leq}$. The **sort of a word** $\alpha \in \Sigma^{\star}_{\leq}$ is the sort of the last symbol appearing in α. Let S_i be the set of all words of sort i. We call any subset of $\Sigma^{\star}_{\leq}$ a **many-sorted language.** We introduce the **many sorted algebra**

$$\mathcal{F} = (S_1, S_2, \ldots, S_k, f_\sigma)_{\sigma \in \Sigma}),$$

where the unary operation f_σ is defined on α and equal to $\alpha\sigma$ if and only if the sort of α is weaker than the sort of σ. We can also define the **many-sorted semigroup**

$$(S_1, S_2, \ldots, S_k, \cdot),$$

where the binary operation $\cdot$ is defined as follows. Let $\alpha, \beta \in \Sigma^{\star}_{\leq}$. Then $\alpha \cdot \beta$ is defined and equal to $\alpha\beta$ if and only if $\alpha\beta \in \Sigma^{\star}_{\leq}$, that is, the sort of α is weaker than the sort of the first letter of β.

Definition 3.5 *A* **many-sorted (non-deterministic) finite automaton** *over the alphabet $\Sigma_{\leq}$ is a system $(Q_1, \ldots, Q_k, I, \Delta, F)$, is defined as follows.*

1. *Each Q_i is the finite* **set of states of sort** *i and $I \subset Q_1$ is the* **set of initial states,**
2. *For all distinct i, j, $Q_i \cap Q_j = \emptyset$.*
3. *$F \subset Q_1 \cup \ldots \cup Q_k$ is the set of* **final states,**
4. *$\Delta \subset (Q_1 \cup \ldots \cup Q_k) \times \Sigma \times (Q_1 \cup \ldots \cup Q_k)$ is the* **transition table** *with the following property. If $(q, \sigma, q') \in \Delta$, then $s(q) \leq s(\sigma) \leq s(q')$.*

If the condition $(q, \sigma, q') \in \Delta$, $(q, \sigma, q'') \in \Delta$ implies that $q' = q''$, then Ω is called **deterministic.**

We can define the notion of computation of many sorted finite automata on sorted words. For a many sorted finite automaton Ω, we can define the set $L(\Omega)$ of all sorted words accepted by the automaton. By the definition of many-sorted automaton, a word accepted by the many-sorted automaton must belong to $\Sigma^{\star}_{\leq}$. We call the set $L(\Omega)$ **recognizable by the many-sorted automaton.** The following lemma can be proved using the standard methods of finite automata theory.

Lemma 3.2 *The set of many-sorted finite automaton recognizable sets is closed under intersection and union.*□

Though the next lemma also uses the known methods of finite automata theory, we present a brief proof of the lemma.

Lemma 3.3 *For any many-sorted finite automaton Ω_1, there exists a deterministic many-sorted finite automaton accepting the same language accepted by Ω_1.*

Proof. Let $\Omega_1 = (Q_1, \dots, Q_k, I, \Delta, F)$. Define the following many-sorted finite automaton Ω_2:

1. For each i the states of sort i are the subsets of Q_i.
2. The set of initial states contatins only one element which is I.
3. The set of final states consists of all subsets intersecting F.
4. The transition table contains all such triples (Q, σ, Q') such that Q and Q' are the states of the new automaton and the following holds:
   (a) For any $q \in Q$ there exists $q' \in Q'$ for which $(q, \sigma, q') \in \Delta$.
   (b) For any $q' \in Q'$ there exists $q \in Q$ for which $(q, \sigma, q') \in \Delta$.

Then Ω_2 is a many-sorted deterministic finite automaton and accepts exactly those words accepted by the original automaton Ω_1.□

Lemma 3.4 *For any many-sorted finite automaton Ω_1, there is a many-sorted finite automaton accepting the language $\Sigma_{\leq} \setminus L(\Omega_1)$.*

Proof. By the previous lemma we may assume that $\Omega_1 = (Q_1, \dots, Q_k, I, \Delta, F)$ is deterministic. Thus $\Omega_2 = (Q_1, \dots, Q_k, I, \Delta, F^c)$ accepts the complement of $L(\Omega_1)$.□

As a corollary of the previous lemmas we get the following theorem.

Theorem 3.1 *The set of all many-sorted finite automata recognizable langauges of $\Sigma_{\leq}$ forms Boolean algebra.*□

Let $\mathcal{F} = (S_1, S_2, \dots, S_k, f_\sigma)_{\sigma\in\Sigma}$ be the above defined many-sorted algebra. An equivalence realtion $\eta \subset \Sigma^\star_{\leq}$ is called a **congruence** if it satisfies the following conditions:

1. For all $(\alpha, \beta) \in \eta$, the words α and β have the same sort.
2. For all $\sigma \in \Sigma$ and $(\alpha, \beta) \in \eta$, if $f_\sigma(\alpha)$ is defined, then $(f_\sigma(\alpha), f_\sigma(\beta)) \in \eta$.

If η is a congruence relation on the many-sorted algebra $\mathcal{F}$, then η is also a **right congruence relation** of the many-sorted semigroup $(S_1, \ldots, S_k, \cdot)$. That is, η satisfies the following condition: for all $(\alpha, \beta) \in \eta$ and $u \in \Sigma^\star_{\leq}$, if $\alpha \cdot u$ is defined, then $(\alpha u, \beta u) \in \eta$.

Let L be a many-sorted language over $\Sigma^\star_{\leq}$. Define the equivalnce relation η_L as follows. For all α, $\beta \in \Sigma^\star_{\leq}$, α and β are η_L–equivalent if

1. α and β have the same sort, and
2. for all $u \in \Sigma^\star_{\leq}$ $\alpha \cdot u \in L$ if and only if $\beta \cdot u \in L$.

Then η_L is a congruence relation on the algebra $\mathcal{F}$ (equivalently, is a **right congruence relation** of the many-sorted semigroup $(\Sigma^\star_{\leq}, \cdot)$). If the alphabet has only one sort, this equivalence is Myhill-Nerode equivalence. The version of the Myhill-Nerode theorem below gives a characterization of many-sorted automata recognizable languages.

Theorem 3.2 *Let L be a many-sorted language over the alphabet $\Sigma_{\leq}$. The following conditions are equivalent:*

1. *The language L is recognizable by a many-sorted finite automaton.*
2. *The langauge L is a union of some equivalence classes of a right congruence relation η of finite index.*

Outline of Proof. Let Ω be a many-sorted automaton accepting L. We may suppose that Ω is deterministic. Then the equivalence relation η_L has finite index and is a right congruence relation. Suppose that L is the union of some equivalence classes of a right congruence relation η of finite index. Define a finite many-sorted automaton accepting L as follows. The initial state of the automaton will be the empty word. The states of sort i are the η–equivalence classes of words of sort i. A state q is a final state of the automaton if q is a subset of L. A pair (q, σ, q') belongs to the transition table of the automaton if $q \cdot \sigma$ belongs to the η–equivalence class q'. One can verify that this automaton is many-sorted and accepts L.□

We apply this theorem to characterize automatic structures. Suppose that A is a finite alphabet and that $\diamond \notin A$. Introduce the alphabet

$$\Sigma = (A \cup \{\diamond\})^n \setminus \{\diamond\}^n.$$

Define a binary relation $\leq$ on the alphabet Σ as follows:

$$(\sigma_1, \ldots, \sigma_n) \leq (\sigma'_1, \ldots, \sigma'_n)$$

if, and only if, for all $i \in \{1, \ldots, n\}$, the condition $\sigma_i = \diamond$ implies that $\sigma'_i = \diamond$. It follows that $\leq$ is a pre-partial ordering of the alphabet Σ, giving us a many-sorted finite alphabet $\Sigma_{\leq}$. By the definition of the convolution operation, the convolution $R^\star$ of any relation R of arity n is a many-sorted language over $\Sigma_{\leq}$. The previous result implies:

Theorem 3.3 *Let A be a finite alphabet, and let $(D, P_0^{m_0}, \ldots, P_s^{m_s})$ be a structure with $D \subset A$ and $P_i^{m_i} \subset (A^\star)^{m_i}$, where $i0, \ldots, s$. For each $i \in \{0, \ldots, s\}$ consider the many-sorted alphabet $\Sigma_{\leq, i} = (A \cup \{\diamond\})^{s_i} \setminus \{\diamond\}^{s_i}$. Then the following statements are equivalent:*

1. *The structure $(D, P_0^{m_0}, \ldots, P_s^{m_s})$ is automatic.*
2. *The set D is a finite automaton recognizable and the convolution of each set $P_i^{s_i}$ is recognizable by a many-sorted finite automaton.*
3. *The set D is a finite automaton recognizable and the convolution of each set $P_i^{s_i}$ is a union of some equivalence classes of a right congruence relation η of finite index on the many-sorted semigroup $(S_1, \ldots, S_k, \cdot)$.*□

We apply the discussion above to characterize automatic structures over one letter alphabet $\Sigma = \{1\}$. We identify the set $\Sigma^\star$ with the set ω. For clarity consider structures with domain ω and binary relations, that is, structures of type

$$(\omega, P_0, \ldots, P_s),$$

where each P_i is a binary relation. We introduce the 3-sorted alphabet

$$\Sigma_{\leq} = \{(1,1), (1,\diamond), (\diamond,1)\},$$

where $(1,1) \leq (1,\diamond)$ and $(1,1) \leq (\diamond,1)$. Let

$$\Omega = (Q_1, Q_2, Q_3, q_0, \Delta, F)$$

be a 3–sorted finite deterministic automaton over the alphabet described as follows. The set Q_1 can be thought as a graph which forms a loop. All transitions in Q_1 are labelled by $(1,1)$. Each $s \in Q_1$ forms two disjoint loops $L_s^{(1,\diamond)}$ and $L_s^{(\diamond,1)}$. All transitions in $L_s^{(1,\diamond)}$ are labelled by $(1,\diamond)$ and all transitions in $L_s^{(\diamond,1)}$ are labelled by $(\diamond,1)$. We also can assume that if $s, s' \in Q_1$ are different states, then the loops $L_s^{(1,\diamond)}$, $L_s^{(\diamond,1)}$, $L_{s'}^{(1,\diamond)}$, $L_{s'}^{(\diamond,1)}$ are disjoint.

A set $M \subset \omega$ is **an arithmetic progression** if there exist numbers $n_1 < \ldots < n_k \in \omega$ such that

$$M = \{n_1, \ldots, n_k\} \cup \{n_k t \mid t \in \omega\}.$$

A subset of ω is **automatic** if it is a finite union of arithmetic progressions. It is easy to see that a set is automatic if and only if it is recognizable by a finite automaton over $\{1\}$.

Corollary 3.1 *Let $(\omega, P_0, \ldots, P_s)$ be a structure such that each P_i is a binary relation. This structure is automatic if and only if for each $i \in \{0, \ldots, n\}$ there exist automatic sets*

$$A_1, B_1, \ldots, A_{i_{k_i}}, B_{i_{k_i}}, \; C_1, D_1, \ldots, C_{i_{t_i}}, B_{i_{t_i}},$$

such that

$$P_i = \bigcup_{j=1}^{k_i} \{(x, x+n) \mid x \in A_j, n \in B_j\} \cup \bigcup_{s=1}^{t_i} \{(x+n, x) \mid n \in C_s, x \in D_s\}.\square$$

4 Basic Properties of Automatic Presentations

Features of Decidability. Investigation of automaton recognizable relations over $\Sigma^\star$ suggests investigating the corresponding predicate calculus. Thus, if R_1 and R_2 are automaton recognizable, one can define relations corresponding to the expressions $(R_1 \vee R_2)$, $(R_1 \wedge R_2)$, and $\neg(R_1)$, $\exists x R_1$, and $\forall x(R_1)$. For instance, suppose R_1 is an n-ary relation. Define

$$\begin{aligned}
\exists x_i(R_1) &= \{(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \mid \\
&\qquad (x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n) \in R_1\}, \\
\forall x_i(R_1) &= \{(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \mid \\
&\qquad \forall \alpha\, R(x_1, \ldots, x_{i-1}, \alpha, x_{i+1}, \ldots, x_n)\}.
\end{aligned}$$

When R_1 is a unary relation, then $\exists x(R_1)$ corresponds to $\emptyset$ if $R_1 = \emptyset$, and to $\Sigma^\star$ otherwise. Similarly, $\forall x(R_1)$ corresponds to $\emptyset$ if $R_1 \neq \Sigma^\star$, and to $\Sigma^\star$ otherwise.

The next theorem is a consequence of standard finite automata theory of the middle 1950's..

Theorem 4.4 *1. Let R_1 and R_2 be automata recognizable relations. Then the relations corresponding to the expressions $(R_1 \vee R_2)$, $(R_1 \wedge R_2)$, $\neg(R_1)$, $\exists x(R_1)$, and $\forall x(R_1)$ are also automata recognizable.*

2. The emptiness problem for n-variable automata is decidable uniformly in n.

3. There exists a procedure which, for automata recognizing R_1 and R_2, constructs automata for recognizing the relations corresponding to the expressions $(R_1 \vee R_2)$, $(R_1 \wedge R_2)$, $\neg(R_1)$, $\exists x(R_1)$, and $\forall x(R_1)$. $\square$

This theorem implies several properties of automatic presentations.

Corollary 4.2 *Suppose that a structure $\mathcal{A}$ has an automatic presentation. Then*

1. *There exists an effective procedure which, applied to a first order definition of a relation P on $\mathcal{A}$, yields an algorithm deciding P.*

2. *The first order theory of the structure A is decidable.*

Proof. Let $\nu : D \to A$ be an automatic presentation. Then the atomic relations and the equality relation on this structure are decidable under the presentation ν. By the theorem just stated, there exists an effective procedure which, for any first order definable relation P, produces an algorithm deciding P. It also follows that the first order theory of this structure is decidable.□

Remark. In fact, if $\mathcal{A}$ is automatic, the corollary above can be strengthened. Namely, consider the set of all polynomials over the structure $\mathcal{A}$. (This is the set of all functions of the form
$t(a_1, \ldots, a_k, x, a_{k+1}, \ldots, a_m)$, where x is a variable, T is a term, and $a_1, \ldots, a_m$ are elements of $\mathcal{A}$.) Using the decidability of the monadic second order theory of two successor functions [12], it follows that the first order theory of $\mathcal{A}$ plus the monadic second order theory of polynomials over $\mathcal{A}$ is decidable [20].

Corollary 4.3 *If structure*

$$\mathcal{A} = (A, f_0^{n_0}, \ldots, f_k^{n_k}, P_0^{m_0}, \ldots, P_s^{m_s}, c_0, \ldots, c_t),$$

has an automatic presentation, then there exists an automatic presentation μ : $d \rightarrow a$ which is a 1-1 function.

Proof. Let $\nu : D_1 \rightarrow A$ be an automatic presentation of $\mathcal{A}$. Let $D \subset \{0, 1, \ldots, n-1\}^\star$. Define ordering $\preceq$ on set $\Sigma^\star$ as follows. If the length of α is less than the length of β, then $\alpha \preceq \beta$. If α and β have the same length, then $\alpha \preceq \beta$ if and only if there exist $\gamma, \gamma_1, \gamma_2 \in \Sigma^\star$ such that $\alpha = \gamma i \gamma_1$, $\beta = \gamma j \gamma_2$ and $i < j \leq n-1$. This relation is recognizable by a 2–variable automaton. Moreover $\preceq$ is a linear ordering of $\Sigma^\star$ isomorphic to the natural ordering of ω. Consider the following relation

$$D = \{\alpha \mid \alpha \in D \& \forall \beta(\nu(\alpha) = \nu(\beta) \rightarrow \alpha \preceq \beta\}.$$

By the theorem above, the set R is a finite automaton recognizable. For every $i \leq k$ there exists a n_i–variable automaton which recognize the set

$$\{(\alpha_1, \ldots, \alpha_{n_i}, \alpha_{n_i+1}) \mid \mathcal{A} \models F_i^{n_i+1}(\nu(\alpha_1), \ldots, \nu(\alpha_{n_i}), \nu(\alpha_{n_i+1}))\}.$$

By the theorem above, the set

$$\begin{aligned} L_i^{n_i+1} \quad = \quad & \{(\alpha_1, \ldots, \alpha_{n_i+1}) \in D^{n_i+1} \mid \\ & \qquad \mathcal{A} \models F_i^{n_i+1}(\nu(\alpha_1), \ldots, \nu(\alpha_{n_i}), \nu(\alpha_{n_i+1}))\} \end{aligned}$$

is also recognizable by an (n_i+1)–variable automaton. Similarly, for each $j \leq s$, the set

$$M_i^{m_j} = \{(\alpha_1, \ldots, \alpha_{m_j}) \in D^{m_j} \mid \mathcal{A} \models P_j^{m_j}(\nu(\alpha_1), \ldots, \nu(\alpha_{n_i}))\}$$

is recognizable by an m_j–variable automaton. It follows that the structure

$$(D, L_0^{n_0+1}, \ldots, L_k^{n_k+1}, M_0^{m_0}, \ldots, M_s^{m_s}, c_0, \ldots, c_t)$$

is isomorphic to $\mathcal{A}$. The mapping $\mu : D \rightarrow A$ defined by $\mu(\alpha) = \nu(\alpha)$ is 1-1. □

Using the theorem above, similar to the previous corollary, one can prove that automata presentable structures are closed under direct product and factorizations with respect to 2–variable automata recognizable congruences.

Corollary 4.4 *1. Let $\nu_1 : D_1 \to A_1$ and $\nu_2 : D_2 \to A_2$ be automatic presentations of structures $\mathcal{A}_1$ and $\mathcal{A}_2$ of the same signature. Then the structure $\mathcal{A}_1 \times \mathcal{A}_2$ possesses an automatic presentation.*

2. Let $\nu_1 : D \to A$ be an automatic presentation of $\mathcal{A}_1$. If η is a congruence of $\mathcal{A}$ recognizable by a 2–variable automaton, then the factor structure $\mathcal{A}/\eta$ possesses an automatic presentation. □

Finitely Generated Automatic Structures. Suppose that a structure $\mathcal{A}$ is finitely generated. Let $a_1, \ldots, a_k$ be generators of this structure. Let $f_1^{m_1}, \ldots, f_t^{m_t}$ be all atomic operations of the algebra. We define the following sequence of finite sets generating the structure.

Stage 0. Put $G_0 = \{a_0, \ldots, a_k\}$.
Stage n+1. Suppose that G_t has been defined. Then

$$G_{n+1} = G_n \cup \{f_i^{m_i}(b_1, \ldots, b_{m_i}) \mid i = 1, \ldots, t, b_1, \ldots, b_{m_i} \in G_n\}.$$

Definition 4.6 *Consider the function f defined by $f(n) = card(G_n)$. We call this function the* **grouth level** *of the generators $a_1, \ldots, a_k$.*

Lemma 4.5 *Let $a_1, \ldots, a_k$ be generators of the automatic structure $\mathcal{A}$ over the alphabet Σ of cardinality s. Then there exist $a, b \in \omega$ such that the growth level of the generators does not exceed s^{a+1+bn}.*

Proof. Since $\mathcal{A}$ is automatic, there exist finite automata $\Omega_1, \ldots, \Omega_t$ recognizing the graphs of the operations. Let a be the maximum of the lengths of the generators $a_1, \ldots, a_k$. Let b be an upper bound for the number of states of all automata Ω_i, $i = 1, \ldots, n$. We can prove by induction on n that $card(G_n) \leq s^{a+1+bn}$.

For the base step, if $n = 0$, $card(G_0) \leq s^{a+1}$, since the number of elements of length n does not exceed s^{n+1}.

For the induction step, sppose the conclusion is true for $card(G_n) \leq s^{a+1+bn}$. Let $a = f(b_1, \ldots, b_m)$ for some atomic operation f and $b_1, \ldots, b_m \in G_n$. By inductive hypothesis, the lengths of the b_i do not exceed $a + bn$. It follows that the length of a does not exceed $a + b(n+1)$. Thus G_{n+1} is a subset of the set of all words of length not exceeding $\leq a + b(n+1)$. The cardinality of this set is at most $s^{a+1+b(n+1)}$, as required.□

Corollary 4.5 *Let $\mathcal{A}$ be an automatic structure. For any substructure generated by some elements $a_1, \ldots, a_k$, there exist $s, a, b \in \omega$ such that the growth level of the generators does not exceed s^{a+bn}.*□

Using this lemma one can get many examples of structures which do not have automatic presentations but still possess the features of decidability mentioned above.

Corollary 4.6 *Suppose that f is a function symbol of arity 2 and c is a constant symbol. Then the absolutely free algebra generated by c and f does not have an automatic presentation.*

Proof. The elements of the algebra are the terms defined by the following inductive definition.

1. c is a term.

2. If t_1 and t_2 are terms, then $f(t_1, t_2)$ is a term.

Let T be the set of terms. The free algebra is (T, f). The generator of this algebra is c. The lower bound for the growth level of the generator c is $2^{2^n} - n$. Therefore the absolutely free algebra (T, f) does not have an automatic presentation. □

5 Strongly Automatic Presentations

In this section we give a characterization of strongly automata presentable structures. Let Σ be a finite alphabet, $\diamond \notin \Sigma$ and $n > 1$. Put $\Sigma_\diamond = \Sigma \cup \{\diamond\}$. On the set of $\Sigma_\diamond^\star$ we define a relation $\sim_\diamond$. Let $\alpha, \beta \in \Sigma_\diamond^\star$. Let $\alpha_\diamond$ be the word obtained from α ommiting all occurences of $\diamond$. Define

$$\alpha \sim_\diamond \beta \leftrightarrow \alpha_\diamond = \beta_\diamond .$$

Thus, $\sim_\diamond$ is an equivalence relation on $\Sigma_\diamond^\star$. We can then define an equivalence relation $\sim_\diamond^n$ on the set of all n-tuples of $\Sigma_\diamond^\star$ as follows. Two n-tuples are equivalent if and only if their corresponding components are $\sim_\diamond$-equivalent. By the definition of $\sim_n^\diamond$, that each $\sim_\diamond^n$-equivalence class is represented by some unique n-tuple from the set $(\Sigma^\star)^n$. Thus, there is a natural 1-1 correspondence between the factor set $(\Sigma_\diamond^\star)^n / \sim_\diamond^n$ and $(\Sigma^\star)^n$. Define a binary operation on $(\Sigma^\star)^n$ as follows.

$$(\alpha_1, \ldots, \alpha_n)(\beta_1, \ldots, \beta_n) = (\alpha_1\beta_1, \ldots, \alpha_n\beta_n)$$

This operation is well-defined on $\sim_\diamond^n$-equivalence classes. Let R be a relation on $\Sigma^\star$ of arity n. Define a Myhill-Nerode equivalence relation $\sim_R$ on $(\Sigma^\star)^n$ by:

$$\alpha \sim_R \beta \leftrightarrow \forall u \in (\Sigma^\star)^n (\alpha u \in R \leftrightarrow \beta u \in R)$$

Then $\sim_R$ is an equivalence relation compatible with the right multiplication, that is, if $\alpha \sim_R \beta$, then for all u, $\alpha u \sim_R \beta u$.

Definition 5.7 *An n–variable automaton $\Omega = (S, S_0, \Delta, F)$ on Σ is* **simple** *if:*

1. *For all $s \in S$ and $\alpha = (\alpha_1, \ldots, \alpha_n), \beta = (\beta_1, \ldots, \beta_n) \in (\Sigma_\diamond^\star)^n$, if $\alpha \sim_\diamond^n \beta$, then $\Delta(s, \alpha_1 \star \ldots \star \alpha_n) = \Delta(s, \beta_1 \star \ldots \star \beta_n)$.*

2. *For all $s \in S$, $\sigma = (\sigma_1, \ldots, \sigma_n), \delta = (\delta_1, \ldots, \delta_n) \in (\Sigma_\diamond)^n$, if $\Delta(s, \sigma) = \Delta(s, \delta)$, then $\Delta(s, \sigma) = \Delta(s, (\gamma_1, \ldots, \gamma_n))$, where $\gamma_i \in \{\sigma_i, \delta_i\}$.*

The next theorem characterizes strongly recognizable relations in terms of simple automata, the equivalence $\sim_R$ and finite automata recognizable subsets of $\Sigma^\star$. The **index** of an equivalence relation is the number of equivalence classes.

Theorem 5.5 *Let Σ be an alphabet. Let R be a relation on $\Sigma^\star$ of arity n. The following statments are equivalent:*

1. *R is a strongly n-recognizable.*
2. *R is a union of some classes of an equivalence relation with a finite index and compatible with the right multiplication on $(\Sigma^\star)^n$.*
3. *R is accepted by a simple n-variable automaton over Σ.*
4. *There exists $k \in \omega$ and finite automata recognizable subsets $R_{1i}, \dots, R_{ni}$ of $\Sigma^\star$, where $1 \le i \le k$, such that $R = \bigcup_{i=1}^{k} R_{1i} \times \ldots \times R_{ni}$.*

Proof. 1) $\to$ 2). Let $\Omega = (S, S_0, \Delta, F)$ be an n–variable strong automaton accepting R. Define a congruence relation $\sim$ on $(\Sigma^\star)^n$ as follows:

$$(\alpha_1, \dots, \alpha_n) \sim (\beta_1, \dots, \beta_n)$$

if, and only if, for all $s, q \in S^n$ and for all $i \le n$, there exists a computation of Ω on α_i which begins in state q and ends in state s if and only if there exists a computation of Ω on β_i which begins in q and ends in s. The relation $\sim$ is a congruence of a finite index. Since R is recognizable by Ω, by the definition of $\sim$, R is a union of some $\sim$–equivalence classes.

2) $\to$ 3). Let $\sim$ be a given equivalence compatible with the right multiplication. Suppose that R is a union of some classes of this equivalence relation. Define an n-variable automaton Ω. Let the set of states to be $S = \{\alpha^{\diamond}_{\sim} \mid \alpha \in (\Sigma^\star)^n\}$, where $\alpha^{\diamond}_{\sim}$ is the $\sim^n_{\diamond}$–closure of the $\sim$–equivalence class containing $\alpha \in (\Sigma^\star)^n$; let the set of initial states to be $S_0 = \{(\lambda, \dots, \lambda)^{\diamond}_{\sim}\}$, where λ is the empty word; put $F = \{\alpha_\sim \mid \alpha \in R\}$; define the transition table Δ as follows: for all $\alpha^{\diamond}_{\sim}$ and $\sigma \in (\Sigma_\diamond)^n$ put $\Delta(\alpha^{\diamond}_{\sim}, \sigma) = (\alpha\sigma)^{\diamond}_{\sim}$. This automaton is simple.

3) $\to$ 4). Suppose that R is accepted by a simple n–variable automaton $\mathcal{A}$ over Σ. Let $f_1, \dots, f_k$ be all final states of the automaton $\mathcal{A}$. Let $R(f_i)$ be the set of all n-tuples which transform the initial state q_0 of $\mathcal{A}$ to f_i. Let $R(f_i, 1), \dots, R(f_i, n)$ be the projections of $R(f_i)$ onto corresponding components. Using the definition of simple automaton, we get $R(f_i) = R(f_i, 1) \times \ldots \times R(f_i, n)$. Thus,

$$R = \bigcup_{i=1}^{k} R(f_i, 1) \times \ldots \times R(f_i, n).$$

Note that for all i, j the set $R(f_i, j)$ is a finite automaton recognizable subset of $\Sigma^\star$.

4) $\to$ 1). For any $i, 1 \le i \le k$, there exists a finite automaton

$$\Omega_i = (S_i, q_{i0}, \Delta_i, F_{i1} \cup \ldots \cup F_{in})$$

such that for any $\alpha \in \Sigma^\star$, $\alpha \in R_{ij}$ if and only if there exists a computation of Ω_i which begins in q_{i0} and ends in F_{ij}. We can suppose that for all $i \ne j$, $S_i \bigcap S_j = \emptyset$. Define an automaton $\Omega = (S, S_0, \Delta, F)$ as follows.

1. $S = S_1 \cup \ldots \cup S_n$ and $S_0 = \{q_{10}, \ldots, q_{k0}\}$.
2. $F = \bigcup_{i=0}^{k} F_{i1} \times \ldots \times F_{in}$.
3. $\Delta = \bigcup \Delta_i$.

This automaton is an n–variable strong automaton which accepts R. □

This theorem and the theorem of the previous section allow us to obtain a corollary characterizing structures which possess strongly automatic presentations. We need the following notion.

An n–ary relation P on a set A is called **complete** if there exist subsets $A_{i1}, \ldots, A_{in}$ of A and a number $k \in \omega$ such that $P = \bigcup_{i=1}^{k} A_{i1} \times \ldots \times A_{in}$.

Theorem 5.6 *A structure*

$$\mathcal{A} = (A; P_0^{m_0}, \ldots, P_t^{m_t})$$

has a strongly automatic presentation if and only if each predicate $P_i^{n_i}$ is complete.

Proof. For simplicity suppose that $t = 0$ and $m_0 = n$. If $\mathcal{A}$ has a strongly automatic presentation (L, R) then, by the previous theorem, there are recognizable sets $R_{1i}, \ldots, R_{ni}$ such that $R = \bigcup_i R_{1i} \times \ldots \times R_{ni}$.. Hence $P_0^{m_0}$ is a complete relation.

Conversely, suppose that P is a complete relation on A. Then there exist subsets $A_{i1}, \ldots, A_{in}$ and a number $k \in \omega$ such that $P = \bigcup_i A_{i1} \times \ldots \times A_{in}$. Consider a structure $\mathcal{A}_1 = (A; A_{11}, \ldots, A_{1n}, \ldots, A_{k1}, \ldots, A_{kn})$. Structure $\mathcal{A}$ is isomorphic to

$$(A; \bigcup_i A_{i1} \times \ldots \times A_{in}).$$

By Proposition 2.1, $\mathcal{A}_1$ has a strong automatic presentation

$$(L; R_{11}, \ldots, R_{1n}, \ldots, R_{k1}, \ldots, R_{kn}).$$

Let $R = \bigcup_i R_{i1} \times \ldots \times R_{in}$. By the previous theorem, relation R is accepted by an n-variable strong automaton. Thus the structure $(L; R)$ is isomorphic to $\mathcal{A}$. Hence $\mathcal{A}$ possesses a strongly automatic presentation.□

Corollary 5.7 *The structures $(\omega; \leq)$ and $(\omega; s)$ do not possess strongly automatic presentations.*□

6 Automatic Isomorphism Types

A basic problem for recursive and polynomial time structures is to characterize structures which have the same isomorphism type via recursive or p–time computable isomorphisms. A structure is recursively (polynomial time) categorical if any two recursive (p–time) presentations of the structure are recursively (p–time) isomorphic. Thus in some sense a recursively (p–time) categorical structure is one for which the problem of recursive (p–time) presentations has a unique solution. It looks very hard to find general necessary and sufficient conditions for structures to be recursively (p–time) categorical. The corresponding problem for automatic isomorphism types is easy.

Definition 6.8 *We say that sets $R_1, R_2 \subset \Sigma^\star$ have the same* **automatic isomorphism type** *if there exists a relation f, recognizable by a 2–variable automaton, such that $dom(f) = R_1$, $range(f) = R_2$, and f is a 1-1 function. In this case we say that R_1 and R_2 are* **automatically isomorphic** *via the* **automatic isomorphism** *f.*

For an R we denote by $AI(R)$ the class of all sets automatically isomorphic to R. Note that if $L \in AI(R)$, then L is a finite automaton recognizable. It is obvious that if R_1 is a finite set, then $R_2 \in AI(R_1)$ if and only if $card(R_1) = card(R_2)$. Our next result shows that for any infinite recognizable set there exists in some sense a standard presentation of this set which has the same AI type. We need definitions.

Definition 6.9 *A set $A \subset \Sigma^\star$ is a* **free monoid** *if*

1. *There exists a $B \subset A$ such that $B^\star = A$, and*
2. *for all $b_1, \ldots, b_s, a_1, \ldots, a_k \in B$ if $b_1 \ldots b_s = a_1 \ldots a_n$, then $k = n$ and $a_i = b_i$ for all $i = 1, \ldots, n$.*

Let $A, B \subset \Sigma^\star$. The mulitiplication $A \cdot B$ is **free** if for for all $a_1, a_2 \in A, b_1, b_2 \in B$ if $a_1 \cdot b_1 = a_2 \cdot b_2$, then $a_1 = a_2$ and $b_1 = b_2$.

Using Eilenberg's decomposition theorem for finite automata recognizable sets [5], one can characterize automatic isomorphism types for structures in the language of pure equality.

Theorem 6.7 *Let L be a finite automaton recognizable infinite set. There exist finite automata recognizable pairwise disjoint sets L_{11}, ..., L_{1k_1}, ..., L_{n1}, ..., L_{nk_n} such that:*

1. *For all i, j, $i = 1, \ldots, n$, $j = 1, \ldots, k_i$, the set L_{ij} is a free submonoid.*
2. *For all $i \neq j$, $i \leq n$, the multiplication $L_{i1} \cdot \ldots \cdot L_{ik_i}$ is free.*
3. *$L = (L_{11} \cdot \ldots \cdot L_{1k_1}) \cup \ldots \cup (L_{n1} \cdot \ldots \cdot L_{kn_k}) \in AI(R)$, where $AI(R)$ is the automatic isomorphism type of R.*

That is, every finite automata recognizable set is automatically isomorphic to a finite union of free mulitiplications of free monoids.□

Now we investigate the more general problem of automatic isomorphism types of automatic structures.

Definition 6.10 *Let $\mathcal{A}$ be an automatic structure. An automatic structure $\mathcal{B}$ is* **automatically isomorphic** *to $\mathcal{A}$ if there exists a function f, recognizable by a 2–variable automaton, such that $dom(f) = A$, $range(f) = B$ and f induces an isomorphism between these structures.*

Let $\mathcal{A}$ be automaton presentable structure. The number of automatic isomorphism types of $\mathcal{A}$ we call the **automatic dimension** of $\mathcal{A}$. The structure $\mathcal{A}$ is **automatically categorical**if its automatic dimension is 1.

Theorem 6.8 *The automatic dimension of any automaton presentable structure is either ω or 1. Moreover, such a structure is automatically categorical if and only if its domain is finite.*

Proof. Let B be an automaton recognizable set. Let us consider the sequence $b_0 \preceq b_1 \preceq b_2 \preceq \ldots$ of all elements of the set B, where $\preceq$ is the linear ordering on $\Sigma^\star$ defined in the proof of Corollary 4.2. We define a function $f_B(n) = length(b_n)$.

Lemma 6.6 *Let B, C be automaton recognizable sets. If the sequence $|f_B(n) - f_C(n)|$ is increasing, then B and C do not have the same automatic isomorphism type.*

Proof. Suppose that the lemma is not true and B, C be automaton recognizable sets such that the set $\{|f_B(n) - f_C(n)| \mid n \in \omega\}$ is not bounded. Let g be 1-1 function such that $g(B) = C$. Suppose that g is recognizable by a 2-variable automaton. Let n be such that for any $b \in B$, we have $|length(b) - length(g(b))| \leq n$. We may assume that n is the number of states of the automaton recognizing g. Indeed, otherwise, applying the pumping lemma of finite automata theory, we would contradict the fact that g is 1-1. Since the sequence $|f_B(m) - f_C(m)|$ is not bounded, there is an $s \in \omega$ such that $|f_B(s) - f_C(s)| > n$. There are two cases.

Case 1. Suppose that $f_C(s) - f_B(s) > n$. Then $g(b_i) \notin \{c_s, c_{s+1}, \ldots\}$ for all $i \leq s$. Since g is 1-1, we should have $\{g(b_0), \ldots, g(b_{s-1})\} = \{c_0, \ldots, c_{s-1}\}$. But we also have $g(b_m) \in \{c_0, \ldots, c_{s-1}\}$, a contradiction.

Case 2. Suppose that $f_B(s) - f_C(s) > n$. Then $f(b_m) \notin \{c_0, \ldots, c_s\}$ for all $m \geq s$. Thus g is not 1-1, a contradiction. We proved the lemma.

Let $\mathcal{A}$ be an infinite automatic structure. Let c be a new symbol which does not belong to Σ. From the structure $\mathcal{A} = (A; P_0^{k_0}, \ldots, P_t^{k_t})$ we define a new structure $\mathcal{B}_n$:

1. The universe of $\mathcal{B}_n$ is
$$B_n = \{a_0 c^n a_1 c^n \ldots a_m c^n \mid a_0 \ldots a_m \in A\}$$
(Notice if $n = 0$ then $B_n = A$.)

2. For each predicate $P_i^{k_i}$ we define a predicate $Q_i^{k_i}$ as follows. A tuple
$$(a_{01} c^n a_{11} c^n \ldots a_{m_0 1} c^n, \ldots, a_{0k_i} c^n a_{1k_i} c^n \ldots a_{m_i k_i} c^n)$$

belongs to $Q_i^{k_i}$ if and only if $(a_{01} \dots a_{m_0 1}, \dots, a_{1k_i} \dots a_{m_i k_i})$ belongs to $P_i^{k_i}$.

This defines a structure

$$\mathcal{B}_n = (B_n; Q_0^{k_0}, \dots, Q_t^{k_t}).$$

¿From the construction, $\mathcal{B}_n$ is isomorphic to $\mathcal{A}$. Since $\mathcal{A}$ is an automatic structure, it follows that $\mathcal{B}_n$ is also an automatic structure.

The the sequence $|f_{\mathcal{B}_{n-1}}(m) - f_{\mathcal{B}_n}(m)|$ is increasing for any fixed $n \in \omega$. The lemma above implies that the structure B_n is not automatically isomorphic to the structure $\mathcal{B}_{n-1}$. To complete the proof, note that finite structures are automatically categorical. □

7 Conclusion and Open Questions

The theory of automatic structures can be considered as a branch of the theory of recursive structures. But one has to take into account the differences between these two approaches for investigating the connections between algebraic, model-theoretic, and effective properties of structures. Recursive model theory can be viewed as an application of recursion theory to model theory, while the theory of automatic structures can be viewed as an application of complexity theory to model theory.

For example, suppose that we have a structure $\mathcal{A}$ and a relation R which is of particular interest. If we are interesting in deciding R, from the recursive structures point of view, we would consider the following type of questions.

1. Does there exist a recursive copy of $\mathcal{A}$ on which R is decidable?

2. Does there exists a recursive copy of $\mathcal{A}$ on which R is creative set?

3. Does there exist a recursive copy of $\mathcal{A}$ on which R is a simple set, or has a particular Turing or m-degree?

But from the point of view of automatic structures, we would naturally consider the following questions.

1. Does there exist an automatic copy of $\mathcal{A}$?

2. Does there exist a automatic presentation of $\mathcal{A}$ on which R is decidable?

3. Does there exist an automatic presentation of $\mathcal{A}$ in which there is a decision procedure for R of given time or space complexity?

4. Does there exist an automatic presentation of $\mathcal{A}$ in which the decision procedure for R is NP–complete?

This paper suggests that recursive, algebraic, model theoretic, and complexity theoretic properties of automatic structures are amendable to systematic investigation. Here are a few open questions.

Question 1. Characterize the first countable ordinal which does not have any automatic presentation.

In section 2, we gave automatic presentations for ordinals ω^n, $n \in \omega$.

Question 2. Give an algebraic or model-theoretic characterization of automata presentable structures.

In section 5 we characterized strongly automata representable structures in terms of complete relations.

Question 3. Characterize decidable first order theories for which every countable model has an automatic reprsnetation.

Section 4 shows that the first order theory of every automatic structure is decidable. On the other hand it is known that every decidable first order theory possesses a decidable structure [7]. It can also be proved that finding the truth value of a fully quantified automaton recognizable predicate is exponential in the size of the automaton recognizing the relation [6].

Question 4. Characterize automatic isomorphism types of automatic structures over a fixed domain.

In the last section, we proved that if we do not fix domains, then any structure with exactly one isomorphism type is finite. We do not know, however, what effect a fixed domain has on automatic isomorphism types.

In this paper we dis not consider presentations of structures using tree automata. One can develop and study tree automata presentatable structures. Of course, an approach based on tree automata would cover this paper and possess the same positive features of decidability. However we decided to present clear definitions and examples based on the simple computational structure of finite automata. The investigation of tree automata presentable structures we have under development. [20].

References

[1] Aspects of Effective Algebra, Proceedings of a Conference at Monash University, 1979, edited by J.N. Crossley.

[2] C.J. Ash, A. Nerode, Intrinsically Recursive Relations, Aspects of Effective Algebra, Proceeding of a Conference at Monash University, Australia, 1979.

[3] R. Buchi, The Monadic Theory of ω_1, in: Decidable Theories II, Lecture Notes in Mathematics, 328, 1-127.

[4] R. Buchi, D. Siefkes, Axiomatization of the Monadic second Order Theory of ω_1, in: Decidable Theories II, Lect. Not. in Mathematics, 328, 127-215.

[5] S. Eilenberg, Automata, Languages, and Mashines, vol. A, Academic Press, New York, 1974.

[6] D.Epstein, J.Cannon, and others, Word Processing in Groups, Jones and Bartlett Publishers, Boston, London.

[7] Yu.L. Ershov, Problems of Decidability and Constructive Models, Moscow, 1989.

[8] A.Frohlich, J. Shepherdson, Effective Procedures in Field Theory, Philos. Tranns. Roy. Soc., London, ser A 248, 1955, 432-487.

[9] S.S. Goncharov, The Problem of the Number Of Non-Self-Equivalent Constructivizations, Algebra and Logic, No 6, 1980.

[10] Y. Gurevich and L. Harrington, Trees, Automata and Games, Proceedings of the 14th Annual ACM Symposium on Theory of Computing, 1982, 60-65.

[11] A.I. Mal'cev A.I, Constructive Algebras, Uspekhi Matem. Nauk, 16, No 3, 1961.

[12] M. Rabin, Decidability of Second-Order Theories and Automata on Infinite Trees, Trans. of American Math. Soc., 141, 1969, 1-35.

[13] M.Rabin, Weakly Definable Relations and Special Automata, Mathematical Logic and Foundations of Set Theory, North-Holland, Amsterdam, 1970, 1-23.

[14] M. Rabin, Computable Algebra: General theory and Theory of Computable Fields, Trans.Amer. Math. Soc., 95, 1960, 341-360

[15] J.B. Remmel, D. Cenzer, Polynomial Time Versus Recursive Models, Annals of Pure and Applied Logic, 1991.

[16] J.B. Remmel, D. Cenzer, Feasibly Categorical Abelian Groups, Proceedings of the Workshop Feasible Mathematics II, to appear.

[17] H. Rogers, Theory of Recursive Function and Effective Computability, New York, 1967.

[18] W. Thomas, Automata on Infinite Objects, in Handbook of Theoretical Computer Science, Volume B, J. van Leeuwen editor, The MIT Press/Elsevier, 1990, 133-192.

[19] C. Frougny, J. Sakarovitch, Synchronized Ratinal Relations of Finite and Infinite Words, Theoretical Computer Science, 108, 1993, 45-82.

[20] B.Khoussainov, A. Nerode, Automatic Models and $S2S$, in preparation.