Subalgebras of FA-presentable algebras

Alan J. Cain and Nik Ruškuc

Abstract. Automatic presentations, also called FA-presentations, were introduced to extend finite model theory to infinite structures whilst retaining the solubility of fundamental decision problems. This paper studies FA-presentable algebras. (The word 'algebra' is used in the sense of universal algebra: a set equipped with a collection of operations.) First, an example is given to show that the class of finitely generated FA-presentable algebras is not closed under forming finitely generated subalgebras, even within the class of algebras with only unary operations. In contrast, a finitely generated subalgebra of an FA-presentable algebra with a single unary operation is itself FA-presentable. Furthermore, it is proven that the class of unary FA-presentable algebras is closed under forming finitely generated subalgebras, and that the membership problem for such subalgebras is decidable.

1. Introduction

Automatic presentations, also known as FA-presentations, were introduced by Khoussainov & Nerode [8] to fulfill a need to extend finite model theory to infinite structures while retaining the solubility of interesting decision problems. Informally, an FA-presentation for a relational structure consists of a regular language of abstract representatives for elements of the structure such that the relations of the structure are recognized by finite automata. FA-presentations have been considered for structures such as orders [9, 10, 5], graphs [7], and groups, semigroups, and rings [14, 2, 13].

This paper studies subalgebras of FA-presentable algebras. (The word 'algebra' is used in the sense of universal algebra: a set equipped with a collection of operations.) In the particular case of groups, it was already known that there exists an FA-presentation for the group $\mathbb{Z} \times \mathbb{Z}$ under which the sublanguage of representatives for elements of any non-trivial cyclic subgroups is not regular [12, § 6]. However, such subgroups, like all abelian groups [14, Theorem 3],

Presented by ...

Received ...; accepted in final form ...

2010 *Mathematics Subject Classification*: Primary: 08A30; Secondary: 03D45, 68Q45.

Key words and phrases: automatic presentations, FA-presentations, algebras, finitely generated subalgebras, unary.

The first author's research was funded by the European Regional Development Fund through the programme COMPETE and by the Portuguese Government through the FCT (Fundação para a Ciência e a Tecnologia) under the project PEst-C/MAT/UI0144/2011 and through an FCT Ciência 2008 fellowship. Two visits by the first author to the University of St Andrews, where much of the research described in this paper was carried out, were supported by the EPSRC-funded project EP/H011978/1 'Automata, Languages, Decidability in Algebra'. The authors thank Richard M. Thomas for helpful discussions.

are FA-presentable with a different language of representatives. By constructing an example of a finitely generated FA-presentable algebra that contains a non-FA-presentable finitely generated subalgebra, we show that the class of FA-presentable algebras is not closed under taking finitely generated subalgebras (Theorem 3.3). Furthermore, this non-closure holds even within the class of algebras equipped with only unary operations (Remark 3.7). However, the class of FA-presentable algebras with a single unary operation is closed under forming finitely generated subalgebras (Proposition 3.2).

On the other hand, we prove that the class of algebras that admit unary FA-presentations (that is, FA-presentations over a one-letter alphabet) *is* closed under forming finitely generated subalgebras (Theorem 5.2). The proof depends on the sublanguage of representatives for elements of the subalgebra being regular and effectively constructable (Theorem 5.1), which also implies that the membership problem is decidable for such subalgebras (Theorem 5.3). We also prove that finitely generated unary FA-presentable algebras have growth level bounded by a linear function (Proposition 5.5). These results fit the general pattern of unary FA-presentations having nice closure and decision properties; see for instance [1, Ch. 7] and [11].

These results for unary FA-presentations are proved using a new diagrammatic representation, developed in § 4. This representation allows us to visualize and manipulate elements of a unary FA-presentable relational structure in a way that is more accessible than the corresponding arguments using languages and automata. In a forthcoming paper [3], we deploy this representation in an analysis of unary FA-presentable binary relations. This representation is thus potentially a unifying framework in which to reason about unary FA-presentable algebraic and relational structures.

2. Preliminaries

The reader is assumed to be familiar with the theory of finite automata and regular languages; see [6, Chs 2–3] for background reading. The empty word (over any alphabet) is denoted ε. Alphabets are always finite.

Throughout the paper, functions are written on the right: $x\phi$ is the result of applying the function ϕ to x.

The set of positive integers $\{1, 2, 3, \ldots\}$ is denoted $\mathbb{N}$, the set of non-negative integers $\{0, 1, 2, \ldots\}$ is denoted $\mathbb{N}^0$.

Definition 2.1. Let L be a regular language over an alphabet A. Define, for $n \in \mathbb{N}$,

$$L^n = \{(w_1, \ldots, w_n) : w_i \in L \text{ for } i = 1, \ldots, n\}.$$

Let \$ be a new symbol not in A. The mapping conv : $(A^*)^n \to ((A \cup \{\$\})^n)^*$ is defined as follows. Suppose

$$\begin{aligned} w_1 &= w_{1,1} w_{1,2} \cdots w_{1,m_1}, \\ w_2 &= w_{2,1} w_{2,2} \cdots w_{2,m_2}, \\ &\vdots \\ w_n &= w_{n,1} w_{n,2} \cdots w_{n,m_n}, \end{aligned}$$

where $w_{i,j} \in A$. Then $\mathrm{conv}(w_1, \ldots, w_n)$ is defined to be

$$(w_{1,1}, w_{2,1}, \ldots, w_{n,1})(w_{1,2}, w_{2,2}, \ldots, w_{n,2}) \cdots (w_{1,m}, w_{2,m}, \ldots, w_{n,m}),$$

where $m = \max\{m_i : i = 1, \ldots, n\}$ and with $w_{i,j} = \$$ whenever $j > m_i$.

Observe that the mapping conv maps an n-tuple of words to a word of n-tuples.

Definition 2.2. Let A be a finite alphabet, and let $R \subseteq (A^*)^n$ be a relation on A^*. Then the relation R is said to be *regular* if

$$\mathrm{conv}R = \{\mathrm{conv}(w_1, \ldots, w_n) : (w_1, \ldots, w_n) \in R\}$$

is a regular language over $(A \cup \{\$\})^n$.

This paper considers algebras as relational structures. An operation of arity n is viewed as a relation of arity $n+1$. For instance, a unary operation (that is, an operation of arity 1, such as inversion in a group) is viewed as a binary relation (that is, a relation of arity 2).

Definition 2.3. Let $\mathcal{S} = (S, R_1, \ldots, R_n)$ be a relational structure. Let L be a regular language over a finite alphabet A, and let $\phi : L \to S$ be a bijective mapping. Then (L, ϕ) is an *automatic presentation* or an *FA-presentation* for $\mathcal{S}$ if, for all relations $R \in \{R_1, \ldots, R_n\}$, the relation

$$\Lambda(R, \phi) = \{(w_1, w_2, \ldots, w_r) \in L^r : R(w_1\phi, \ldots, w_r\phi)\},$$

where r is the arity of R, is regular.

If $\mathcal{S}$ admits an FA-presentation, it is said to be *FA-presentable*.

If (L, ϕ) is an FA-presentation for $\mathcal{S}$ and L is a language over a one-letter alphabet, then (L, ϕ) is a *unary* FA-presentation for $\mathcal{S}$, and $\mathcal{S}$ is said to be *unary FA-presentable*.

The usual definition of an FA-presentation (L, ϕ) allows the map ϕ to be surjective and not necessarily injective and requires $\Lambda(=, \phi)$ to be regular. However, if a relation structure admits an FA-presentation in this more general sense, it admits an FA-presentation in the sense we use, where the map ϕ is bijective [8, Corollary 4.3].

Every FA-presentable structure admits a *binary* FA-presentation; that is, where the language of representatives is over a two-letter alphabet; see [1, Lemma 3.3]. Therefore the class of binary FA-presentable structures is simply the class of FA-presentable structures. However, there are many structures that admit FA-presentations but not unary FA-presentations: for instance, any finitely generated virtually abelian group is FA-presentable [14, Theorem 8], but unary FA-presentable groups must be finite [1, Theorem 7.19]. Furthermore, the two classes have different decidability properties: FA-presentable structure have decidable first-order theory (cf. Proposition 2.4 below), but unary FA-presentable structures have decidable monadic second-order theory. Thus there are fundamental differences between unary FA-presentable structures and all other FA-presentable structures.

The fact that a tuple of elements $(s_1, \ldots, s_n)$ of a structure $\mathcal{S}$ satisfies a first-order formula $\theta(x_1, \ldots, x_n)$ is denoted $\mathcal{S} \models \theta(s_1, \ldots, s_n)$.

Proposition 2.4 ([8, Theorem 4.4])**.** *Let $\mathcal{S}$ be a structure with an FA-presentation (L, ϕ). For every first-order formula with parameters $\theta(x_1, \ldots, x_n)$ using relation symbols from the structure and equality, the relation*

$$\Lambda(\theta, \phi) = \big\{(w_1, \ldots, w_n) \in L^n : \mathcal{S} \models \theta(w_1\phi, \ldots, w_n\phi)\big\}$$

is regular, and an automaton recognizing it can be effectively constructed.

Proposition 2.4 is fundamental to the theory of FA-presentations and will be used without explicit reference throughout the paper.

The following important result shows that in the case of unary FA-presentations for infinite structures, we can assume that the language of representatives is the language of *all* words over a one letter alphabet:

Theorem 2.5 ([4, Theorem 3.1])**.** *Let $\mathcal{S}$ be an infinite relational structure that admits a unary FA-presentation. Then $\mathcal{S}$ has an unary FA-presentation (a^*, ψ).*

The proof of Theorem 2.5 relies on the essentially 'linear' nature of the set a^*: the the idea is that if we have a unary FA-presentation (L, ϕ) with $L \subseteq b^*$, then in the new FA-presentation (a^*, ψ), the word a^i represents the same element as i-th word of L in increasing order of length. This idea has no direct extension to the case of non-unary alphabets.

We reiterate that 'unary' is used here in two ways: a unary operation of an algebra is an operation of arity 1 (viewed as a relation of arity 2), and a unary FA-presentation is an FA-presentation over a one-letter alphabet.

Definition 2.6. Let $\{S_i : i \in I\}$ be a set of semigroups. The *zero-union* of the S_i is the set $\{0_S\} \cup \bigcup_{i \in I} S_i$, where 0_S is a new element, with multiplication defined as follows: if $s, t \in S_i$, then their product $st \in S_i$ is as before; otherwise their product is 0_S. This multiplication is associative and so the zero-union of the S_i is itself a semigroup.

Definition 2.7. Let A be an alphabet equipped with some total order $\leq$. Let $L \subseteq A^*$. The *length-plus-lexicographic* ordering of words in L induced by $\leq$, denoted by $\sqsubseteq$, is the total order

$$x_1 x_2 \cdots x_k \sqsubseteq y_1 y_2 \cdots y_l \iff k < l \vee \Big(k = l \wedge (\exists i)\big(x_i \leq y_i \wedge (\forall j < i)(x_j = y_j)\big)\Big).$$

That is, $\sqsubseteq$ first orders words by length and then orders words of the same length lexicographically with respect to $\leq$.

3. Subalgebras of FA-presentable algebras

This section presents various contrasting results for finitely generated subalgebras of FA-presentable subalgebras. After a preliminary discussion of algebras equipped with a single unary operation, we prove that the class of FA-presentable algebras is not closed under taking finitely generated subalgebras (Theorem 3.3), by exhibiting an example of a finitely generated FA-presentable algebra that admits a non-FA-presentable finitely generated subalgebra. Although this example algebra is equipped with a binary operation, we note afterwards how it can be modified into an algebra with only unary operations (Remark 3.7), which contrasts the situation for an algebra with a single unary operation.

Although this section shows that the class of FA-presentable algebras is not closed under forming finitely generated subalgebras, closure under forming finitely generated subalgebras may hold within classes of FA-presentable algebras of a particular type. For instance, the following result holds:

Proposition 3.1. *Any finitely generated subgroup of an FA-presentable group FA-presentable.*

Proof. A finitely generated subgroup of an FA-presentable group is virtually abelian by [13, Theorem 10(i)] and hence FA-presentable by [14, Theorem 3]. □

In the case of algebras whose signature comprises a single unary operation, a positive result holds again: the class of FA-presentable such algebras is closed under forming FA-presentable subalgebras. This follows *a fortiori* from the following stronger result:

Proposition 3.2 (Kuske [Personal communication])**.** *Let $\mathcal{S} = (S, \sigma)$ be a finitely generated algebra with a single unary operation σ. Then $\mathcal{S}$ is unary FA-presentable.*

Proof. The algebra $\mathcal{S}$ is simply a set with a single unary operation. Thus we can view $\mathcal{S}$ as a directed graph with vertex set S and edge relation σ. It is easy to see that this graph consists of a finite graph (consisting of the elements x such that $x\sigma^k = x\sigma^{k+l}$ for some $k, l \in \mathbb{N}$) with finitely many infinite rays attached (consisting of the elements x such that elements $x\sigma^k$ are distinct for all $k \in \mathbb{N}$). Suppose that the finite part contains m elements and that there are n such infinite rays. Let $\phi : a^* \to S$ map the words $a^0, \ldots, a^{m-1}$ to the m elements of the finite part and map a^{m+ni+j} (where $i \in \mathbb{N}^0$ and $0 \leq j < n$) to the i-th element of the j-th infinite ray. Then $\Lambda(\sigma, \phi)$ is the union of a finite relation and

$$\{(a^{m+ni+j}, a^{m+n(i+1)+j}) : i \in \mathbb{N}^0, 0 \leq j < n\} = (a, a)^*(a^m, a^{m+n});$$

thus $\Lambda(\sigma, \phi)$ is regular. Thus (a^*, ϕ) is a unary FA-presentation for $\mathcal{S}$. □

Note, however, that Proposition 3.2 does not yield an effective algorithm for constructing the subalgebra. If such an algorithm existed, reachability in the configuration graphs of deterministic Turing machines would be soluble.

Theorem 3.3. *The class of FA-presentable algebras is not closed under taking finitely generated subalgebras.*

Proof. We will construct an FA-presentable $\mathcal{X}$ and show it contains a finitely generated non FA-presentable subalgebra. The example algebra $\mathcal{X}$ will consist of the disjoint union of a semilattice and two copies of the configuration graph of a Turing machine, augmented by extra unary operations.

For each $i \in \mathbb{N}$, let M_i be a chain of 2^i elements. Let S be the zero-union of all the M_i; the zero of S is denoted 0_S. Notice that S is a semilattice and can be viewed either as a partially-ordered set or as a semigroup where the multiplication is the meet operation.

Let $\mathfrak{T}$ be a deterministic Turing machine that generates sequences of symbols a^{j^2}, where $j \in \mathbb{N}$. More precisely, $\mathfrak{T}$ starts with an empty tape, performs some computation and arrives in a distinguished state $q_\square$ with its tape contents being a^{1^2}, then computes again and reaches state $q_\square$ with its contents being a^{2^2}. In general at various points during its computation $\mathfrak{T}$ has tape contents a^{j^2} for every $j \in \mathbb{N}$, and $\mathfrak{T}$ enters state $q_\square$ exactly when its tape contents are a^{j^2} for some $j \in \mathbb{N}$. Notice that $\mathfrak{T}$ runs forever without halting. Suppose Q is the state set and B the tape alphabet of $\mathfrak{T}$.

Recall that an *instantaneous description*, or *configuration*, of $\mathfrak{T}$ consists of its state, its tape contents, and the position of its read/write head on its tape. The *configuration graph* of $\mathfrak{T}$ is an infinite graph whose vertices are all conceivable configurations of $\mathfrak{T}$, with a directed edge from g to g' precisely if $\mathfrak{T}$, when in configuration g, can make a single computation step and reach configuration g'. Note that in general not all configurations are reachable from the initial configuration.

Let $G_\top$ and $G_\bot$ be two copies of the configuration graph of $\mathfrak{T}$. The carrier set for the algebra $\mathcal{X}$ will be $X = S \cup G_\top \cup G_\bot$. The semilattice S is already equipped with a multiplication $\circ$; extend this multiplication to X by defining $g \circ g' = g$ and $g \circ s = s \circ g = g$ for $g, g' \in G_\top \cup G_\bot$ and $s \in S$. The configuration graph $G_\bot$ is equipped with a directed edge relation δ. Since $\mathfrak{T}$ is deterministic, each vertex of the graph has outdegree 1, and so the relation δ can be viewed as a unary operation. Extend δ to X by $x\delta = x$ for all $x \in S \cup G_\top$. We emphasize that δ acts like a computation step by $\mathfrak{T}$ in the configuration graph $G_\perp$, but acts like the identity map on the configuration graph $G_\top$.

Now define three new unary operations. First, α sends each configuration in $G_\top$ to the corresponding configuration in $G_\perp$, and otherwise (for all elements of $S \cup G_\perp$) acts like the identity map. Second, β sends each element of a chain M_i to the element immediately below it in that same M_i, sends the minimum element of each M_i to 0_S, and otherwise (for all elements of $\{0_S\} \cup G_\top \cup G_\perp$) acts like the identity map. Third, γ maps configurations in $G_\perp$ with state $q_\Box$ and tape contents a^k to the maximum element of the chain M_k, and otherwise (for all other elements of $G_\perp$ and all elements of $S \cup G_\top$) acts like the identity map.

So far, we have a set X equipped with operations $\circ$, α, β, γ, and δ. We still have to define one more unary operation ζ before we obtain our example algebra $\mathcal{X} = (X, \circ, \alpha, \beta, \gamma, \delta, \zeta)$. However, we must first start to define an FA-presentation (L, ϕ) for $\mathcal{X}$, because we will define ζ in terms of the representatives in L.

Let L be the language $\{z\} \cup \{0,1\}^* \cup \{\top, \perp\}B^*QB^*$, where z is a new symbol not in B or Q. Define $\phi : L \to X$ as follows:

- $z\phi = 0_S$.
- If $u \in \{0,1\}^k$, then $u\phi$ is the u-th element (interpreting u as a binary number) from the bottom in M_k. (Notice that since M_k contains exactly 2^k elements, ϕ restricts to a bijection between $\{0,1\}^k$ and M_k.)
- If $t \in \{\top, \perp\}$, $u, v \in B^*$ and $q \in Q$, then $(tuqv)\phi$ is the configuration in G_t where the state is q, the tape contains uv, and the head points to the first symbol in v.

Let us first show that the definition of FA-presentability is satisfied for the operations $\circ$, α, β, γ, and δ.

To see that $\Lambda(\circ, \phi)$ is regular, it is simplest to notice that the $\Lambda(\leq, \phi)$ is regular, where $\leq$ is the order on the semilattice S. This is true because an automaton recognizing $\Lambda(\leq, \phi)$ must simply perform two tasks: (1) compare the lengths of two strings over $\{0,1\}^*$ and then compare them as binary numbers, and (2) always accept if the left-hand input word is z and the right lies in $\{z\} \cup \{0,1\}^*$. The operation $\circ$ is first-order definable in terms of $\leq$, since

$$\begin{aligned} s \circ t = x \iff & (x \leq s) \wedge (x \leq t) \\ & \wedge (\forall y \in S)\big(\big((y \leq s) \wedge (y \leq t)\big) \implies (y \leq x)\big). \end{aligned}$$

Thus $\Lambda(\circ, \phi)$ is regular.

Next,

$$\begin{aligned} \Lambda(\alpha, \phi) = & \{(\top uqv, \perp uqv) : u, v \in B^*, q \in Q\} \\ & \cup \{(w, w) : w \in \{z\} \cup \{0,1\}^* \cup \perp B^*QB^*\} \end{aligned}$$

is clearly regular.

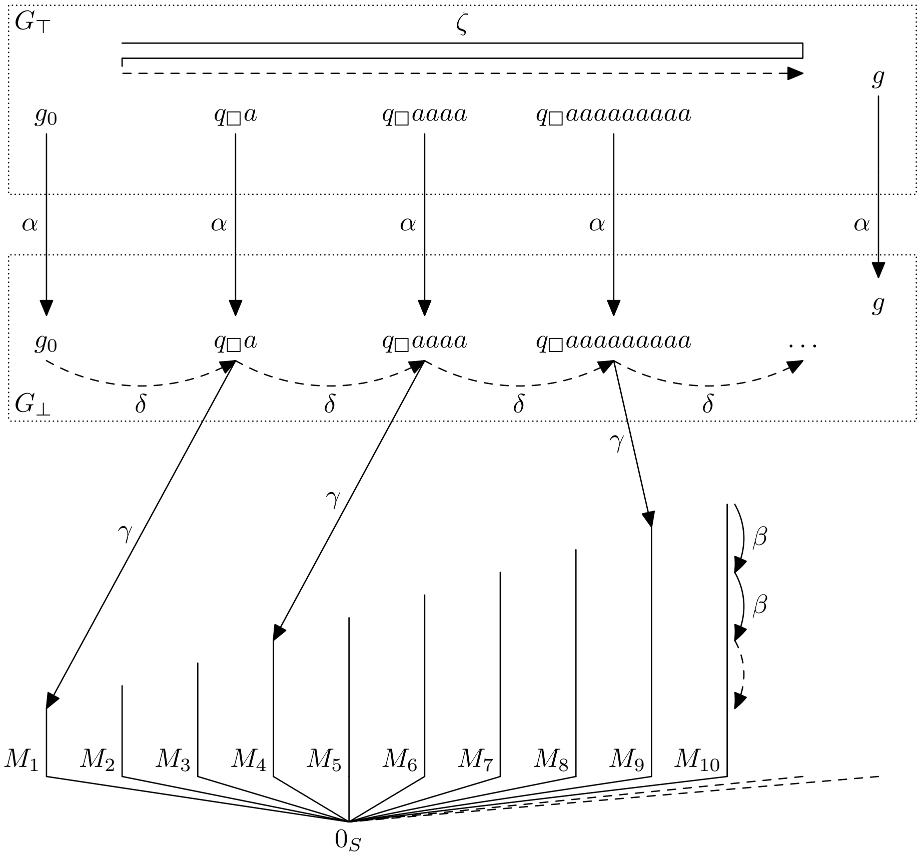


Figure 1. Schematic diagram of the algebra $\mathcal{X}$ from the proof of Theorem 3.3 . The unary operation α maps each configuration of $G_\top$ to the corresponding configuration of $G_\perp$; the operation β maps 'down' within the semilattice; the operation γ maps from '$q_\square$' configurations of $G_\perp$ to the maximum element of M_i, where i is the number of symbols a on the tape; the operation δ mimics the computation step of the Turing machine $\mathfrak{T}$; and the operation ζ iterates through $G_\top$ in length-plus-lexicographic order.

An automaton recognizing $\Lambda(\beta, \phi)$ need only decrement a binary number by 1, recognize $(0^k, z)$, and recognize the identity relation on $\{z\} \cup \{\top, \perp\} B^* Q B^*$.

Now,

$$\Lambda(\gamma, \phi) = \{(\perp a^k q_\square a^l, 1^{k+l}) : k, l \in \mathbb{N}^0\} \cup \{(u, u) : u \in L - \perp a^* q_\square a^*\},$$

which is easily seen to be regular.

The relation $\Lambda(\delta, \phi)$ is easily seen to be regular, since each computation of a Turing machine makes only a small localized change to the configuration as represented by words in B^*QB^*; see [8, p. 374] and [15, Example B.2.19].

We can now define our last operation ζ. Let $\sqsubseteq$ be the length-plus-lexicographic ordering of words in $\top B^*QB^*$ induced by some order on $\{\top\} \cup B \cup Q$. For any element $g \in G_\top$, define $g\zeta$ as follows. Let u be the unique word in $\top B^*QB^*$ with $u\phi = g$. Let u' be the word in $\top B^*QB^*$ that succeeds u in the $\sqsubset$ ordering. Then $g\zeta$ is defined to be $u'\phi$. For all $x \in S \cup G_\perp$, define $x\zeta = x$. Notice that

$$\begin{aligned}\Lambda(\zeta, \phi) = \big\{(u, u') : u \in \top B^*QB^* \wedge (u \sqsubset u') \\ \wedge (\forall v \in \top B^*QB^*)(u \sqsubset v \implies u' \sqsubseteq v)\big\} \\ \cup \{(w, w) : w \in L - \top B^*QB^*\}\end{aligned}$$

is regular since a synchronous automaton can recognize the $\sqsubset$ relation [15, Example B.1.10].

Thus (L, ϕ) is an FA-presentation for the algebra $\mathcal{X} = (X, \circ, \alpha, \beta, \gamma, \delta, \zeta)$.

Lemma 3.4. *The algebra $\mathcal{X}$ is finitely generated.*

Proof. Let u be the $\sqsubseteq$-minimal word in $\top B^*QB^*$. Recall that $u\phi$ is the element of X represented by u; thus $u\phi \in G_\top$. Let T be the set of elements in the subalgebra generated by $u\phi \in G_\top$; the aim is to show that $T = X$.

By repeated application of the operation ζ to $u\phi$, all elements of $G_\top$ lie in T. By applying α to elements of $G_\top$, all elements of $G_\perp$ lie in T. By applying γ to those configurations in $G_\perp$ where the state is $q_\Box$ and the tape contains a^k for some $k \in \mathbb{N}$, the maximum elements of each chain M_i lie in T. By repeatedly applying β to these maximum elements, all elements of the chains M_i lie in T, as does 0_S. Hence all elements of X lie in T and so $X = T$. $\square$

Let g_0 be the initial configuration of $\mathfrak{T}$ in the configuration graph $G_\perp$. Let $\mathcal{Y}$ be the subalgebra generated by g_0. In order to prove that $\mathcal{Y}$ is not FA-presentable, we require the following combinatorial lemma:

Lemma 3.5. *Let $\vartheta : \mathbb{N}^0 \to \mathbb{N}^0$ be an injection. Then there are infinitely many $i \in \mathbb{N}$ such that $i \leq i\vartheta$.*

Proof. Suppose, with the aim of obtaining a contradiction, that there are only finitely many $i \in \mathbb{N}^0$ such that $i \leq i\vartheta$. Let $I = \{i \in \mathbb{N} : i \leq i\vartheta\}$; by supposition, I is finite. Let $m = \max(I)$ and $n = \max(I\vartheta)$. Then $m \leq m\vartheta$ and $m\vartheta \leq n$, so $m \leq n$. Furthermore, $i\vartheta < i$ for $i \notin I$, and $i\vartheta \leq n$ for $i \in I$. Hence $i\vartheta \leq n$ for all $i \leq n$. Since $m = \max(I)$ and $m \leq n$, it follows that $n + 1 \notin I$ and so $(n + 1)\vartheta < n + 1$. Putting the last two sentences together shows that $\{0, \ldots, n + 1\}\vartheta \subseteq \{0, \ldots, n\}$, which contradicts ϑ being an injection. Thus there are infinitely many $i \in \mathbb{N}^0$ such that $i\vartheta \geq i$. $\square$

Lemma 3.6. *The subalgebra $\mathcal{Y}$ is not FA-presentable.*

Proof. The first step is to show that the subalgebra $\mathcal{Y}$ contains the chains M_{j^2} and no other chains M_i.

Recall that $\mathcal{Y}$ is generated by g_0, the initial configuration of $\mathfrak{T}$ in $G_\perp$. The operation δ applied repeatedly to g_0 yields every element reachable from g_0 in the configuration graph $G_\perp$. Let H be the set of these reachable elements. By the definition of $\mathfrak{T}$, the set H includes configurations with state $q_\Box$ and tape contents a^{j^2} for all $j \in \mathbb{N}$. Furthermore, the definition of $\mathfrak{T}$ ensures that H includes no other configuration with state $q_\Box$. The operation γ applied to H yields the maximum element of every M_{j^2} (where $j \in \mathbb{N}$). The operation β now yields all elements of each M_{j^2} and also yields 0_S. So $\mathcal{Y}$ contains the set $Y = \{0_S\} \cup H \cup \bigcup_{j\in\mathbb{N}} M_{j^2}$. It is easy to see that Y is closed under every operation. So the domain of $\mathcal{Y}$ is Y. In particular, $\mathcal{Y}$ contains the chains M_{j^2} and no other chains M_i.

Now suppose, with the aim of obtaining a contradiction, that $\mathcal{Y}$ admits an automatic presentation (L, ϕ). Let

$$Y_1 = \{y \in Y : y \circ 0_S = 0_S\}; \tag{3.1}$$

notice that $Y_1 = \{0_S\} \cup \bigcup_{j\in\mathbb{N}} M_{j^2}$. Notice further that Y_1 is defined by a first order formula and so $L_1 = Y_1\phi^{-1}$ is regular. Observe that the order relation $\leq$ on the subsemilattice Y_1 is first-order definable in terms of $\circ$. Let

$$K_1 = \big\{u \in L_1 : (\forall v)\big((u\phi \leq v\phi) \implies (u\phi = v\phi)\big)\big\}. \tag{3.2}$$

Then K_1 consists of representatives in L of the maximum elements in the various sub-chains M_{j^2} of Y_1. Since it is defined by a first-order formula, K_1 is regular. Let

$$K_2 = \big\{u \in K_1 : (\forall v \in L_1)\big((v \in K_1 \wedge |u| = |v|) \implies (v \sqsubseteq u)\big)\big\};$$

then K_2 consists of length-plus-lexicographically minimal words of each length in K_1. The language K_2 is regular. The relation

$$R_1 = \big\{(u, v) : (u \in K_2) \wedge (u\phi \geq v\phi) \wedge (v\phi \neq 0_S)\big\} \tag{3.3}$$

is regular. Notice that R_1 relates a word $u \in K_2$, which represents the maximum element of some chain M_{j^2}, to all the words v representing elements of that chain. Let n be the number of states in an automaton recognizing $\operatorname{conv}(R_1)$.

If $(u, v) \in R_1$, then $|v| \leq |u| + n$, for otherwise one could pump the subword of v that extends beyond u to obtain infinitely many words representing elements of a single M_{j^2}, which would entail infinitely many distinct elements of M_{j^2} (since ϕ is injective), which is a contradiction.

Let

$$R_2 = \{(u\#^n, v) : (u, v) \in R_1\},$$

where $\#$ is a new symbol. By the observation in the last paragraph, if $(u, v) \in R_2$, then $|u| \geq |v|$. Furthermore, if $(u, v), (u', v') \in R_2$ and $|u| = |u'|$, then $u = u'$ by the definition of R_2 and K_2. Moreover, no word in $\operatorname{conv}(R_2)$

contains a letter whose left-hand component is \$. Therefore the number of words of length k in $\mathrm{conv}(R_2)$ is either 0 or, if there is a word $u \in K_2$ of length $k-n$, the number of possible words v such that $(u\#^n, v)$ lies in R_2, which is in turn the number of elements of the chain M_{j^2} in which $u\phi$ lies, which is 2^{j^2}.

Let z_k be the number of words in $\mathrm{conv}(R_2)$ of length k. By the observation in the last paragraph, whenever z_k is non-zero, it is the number of elements in some chain M_{j^2}. Since $\mathrm{conv}(R_2)$ is a regular language, the generating function

$$f(x) = \sum_{k=0}^{\infty} z_k x^k$$

is a rational function with no singularity at 0. Thus the radius of convergence of its power series expansion must be strictly greater than zero. The aim is to obtain a contradiction by showing that this power series has radius of convergence zero.

By the pumping lemma for regular languages, there are constants p, q such that z_{p+kq} is non-zero for all $k \in \mathbb{N}^0$. So for every $k \in \mathbb{N}^0$, there exists $k\vartheta \in \mathbb{N}^0$ such that $z_{p+kq} = 2^{(k\vartheta)^2}$. This defines an injection $\vartheta : \mathbb{N}^0 \to \mathbb{N}^0$. By Lemma 3.5, $k \leq k\vartheta$ for infinitely many values of $k \in \mathbb{N}^0$. So by choosing k to be large enough and also satisfying $k \leq k\vartheta$, the value

$$|z_{p+kq}|^{1/(p+kq)} = \left|2^{(k\vartheta)^2}\right|^{1/(p+kq)} = \left|2^{(k\vartheta)^2/(p+kq)}\right|$$

can be made arbitrarily large. Therefore

$$\limsup_{k\to\infty} |z_{p+kq}|^{1/(p+kq)} = \infty,$$

and hence $\limsup_{k\to\infty} |z_k|^{1/k} = \infty$, from which it follows that the radius of convergence of the power series $\sum_{k=0}^{\infty} z_k x^k$ is zero. $\square$

By Lemma 3.4, the algebra $\mathcal{X}$ is finitely generated, but contains the finitely generated subalgebra $\mathcal{Y}$, which is not FA-presentable by Lemma 3.6. This completes the proof of Theorem 3.3. $\square$

Remark 3.7. In the proof of Theorem 3.3, the algebra $\mathcal{X}$ has exactly one binary operation, namely the multiplication $\circ$. However, this is not used for the finite generation of $\mathcal{X}$ or $\mathcal{Y}$, and is used in only three places in the proof of Lemma 3.6, namely (3.1), (3.2), and (3.3). (In the latter two, it is hidden within the first-order definition of the order $\leq$ on S.)

However, we can modify $\mathcal{X}$ by removing $\circ$ and adding two new unary operations λ and μ to obtain a new algebra $\mathcal{X}'$, where the subalgebra $\mathcal{Y}'$ generated by g_0 has the same domain Y as $\mathcal{Y}$ and where the $\mathcal{Y}'$ can be proved to be non-FA-presentable in the same way. Hence, even the class of FA-presentable algebras with only unary operations is not closed under forming finitely generated subalgebras.

The first operation λ sends every element of each chain M_i to the maximum element of that chain, and acts like the identity map elsewhere (that is, on

$\{0_S\} \cup G_\top \cup G_\perp)$. The second operation μ sends every element of S to 0_S and acts like the identity map elsewhere (on $G_\top \cup G_\perp$). Notice that

$$\Lambda(\lambda, \phi) = \{(u, 1^{|u|}) : u \in \{0,1\}^*\} \cup \{(u, u) : u \in \{z\} \cup \{\top, \perp\}B^*QB^*\}$$
$$\Lambda(\mu, \phi) = \{(u, z) : u \in \{z\} \cup \{0,1\}^*\} \cup \{(u, u) : u \in \{\top, \perp\}B^*QB^*\};$$

both relations are clearly regular. Thus $\mathcal{X}' = (X, \alpha, \beta, \gamma, \delta, \zeta, \lambda, \mu)$ is FA-presentable and has only unary operations.

To prove that the subalgebra $\mathcal{Y}'$ (generated by g_0) is not FA-presentable, follow the proof of Lemma 3.6, with the following definitions for Y', K_1, and R_1 replacing (3.1), (3.2), and (3.3):

$$\begin{aligned} Y_1 &= \{y \in Y : y\mu = 0_S\}, \\ K_1 &= \big\{u \in L_1 : ((u\phi)\lambda = u\phi) \wedge (u\phi \neq 0_S)\big\}, \\ R_1 &= \big\{(u, v) : (u \in K_2) \wedge (u\phi = (v\phi)\lambda)\big\}. \end{aligned}$$

Note that these are first order definitions in terms of the new signature in which λ and μ replace $\circ$.

Remark 3.8. An anonymous referee of a previous version of this paper pointed out that the proof of Theorem 3.3 can be adapted to give an example of a subalgebra that does not have solvable first order theory (and so cannot be FA-presentable by Proposition 2.4). The idea is as follows. Let $K \subseteq \mathbb{N}$ be recursively enumerable but not recursive. Replace the Turing machine $\mathfrak{T}$ in the proof of Theorem 3.3 with one that enumerates K, in the sense that it enters the state $q_\Box$ exactly when its tape contains a^k for some $k \in K$. Using this modified version of the algebra $\mathcal{X}$, the subalgebra $\mathcal{Y}$ contains exactly the chains M_k for $k \in K$. That is, $k \in K$ if there is a maximal chain (with respect to the relation $\geq$) of k elements $y_1 > y_2 > \ldots > y_k$ not equal to 0_S in the subalgebra $\mathcal{Y}$. Since this condition can be expressed in a first-order formula, it follows from the fact that K is not recursive that $\mathcal{Y}$ cannot have solvable first-order theory.

4. Diagrams for unary FA-presentations

This section develops a diagrammatic representation for unary FA-presentations. In the following section, we apply this representation to prove results about subalgebras of unary FA-presentable algebras.

Let $\mathcal{S}$ be a unary FA-presentable structure with relations $R_1, \ldots, R_n$. Then by Theorem 2.5 it admits a unary FA-presentation (a^*, ϕ). For each $i \in \{1, \ldots, n\}$, let $\mathfrak{A}_i$ be a deterministic r_i-tape automaton recognizing $\Lambda(R_i, \phi)$, where r_i is the arity of R_i. Let us examine the structure of the automata $\mathfrak{A}_i$. For ease of explanation, view $\mathfrak{A}_i$ as a directed graph with no failure states: $\mathfrak{A}_i$ fails if it is in a state and reads a symbol that does not label any outgoing edge from that state.

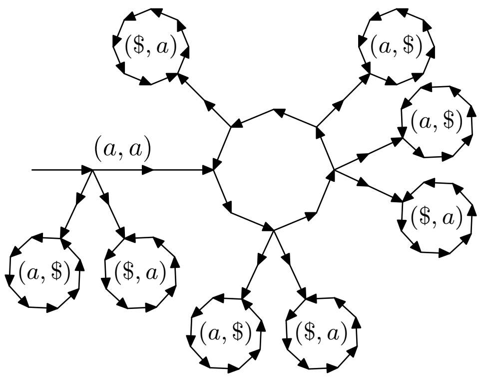


FIGURE 2. Example of an automaton recognizing $\Lambda(R_i, \phi)$ where r_i is 2. Edges labelled (a, a) form a path that leads into a uniquely determined loop. From this path and loop paths labelled by $(a, \$)$ or $(\$, a)$ branch off. (Notice that $(a, \$), (\$, a) \prec (a, a)$.)

Define a partial order $\prec$ on elements of $\{a, \$\}^{r_i}$ as follows: $(x_1, \ldots, x_{r_1}) \preceq (x'_1, \ldots, x'_{r_1})$ if and only if $x'_i = \$ \implies x_i = \$$ for all i. Since $\mathfrak{A}_i$ recognizes words in $\operatorname{conv}((a^*)^{r_i})$, it will only successfully read words consisting of a $\preceq$-decreasing sequence of tuples in $\{a, \$\}^{r_i}$. Thus an edge labelled by a tuple b leads to a state all of whose outgoing edges are labelled by $\preceq$-preceding tuples.

Since $\mathfrak{A}_i$ is deterministic, while it reads letters of a fixed tuple $b \in \{a, \$\}^{r_i}$, it follows a fixed path which, if the string of letters b is long enough, will form a uniquely determined loop. This loop, if it exists, is simple. From various points along this loop and the path leading to it, paths labelled by $\prec$-preceding letters of $\{a, \$\}^{r_i}$ may 'branch off'. Figure 2 shows an example where r_i is 2.

We are going to define a constant D which functions as a 'universal' pumping constant: subject to certain restrictions, we will be able to pump so as to increase or decrease lengths by D. When we define the diagrammatic representation, it is this 'universality' that will allow us to view pumping as translation in the diagram.

Let D_i be a multiple of the lengths of the loops in $\mathfrak{A}_i$ (as discussed above) that also exceeds the number of states in $\mathfrak{A}_i$. Let D be a multiple of the various D_i.

Fix some i. Let $\mathfrak{A}_i$ have initial state q_0 and transition function δ. We extend the transition function in the usual way from single symbols to words. Consider a word $uvw \in \operatorname{conv}(L(\mathfrak{A}_i))$, where $v = b^\beta$ for some $b \in \{a, \$\}^{r_i}$ and $\beta \geq D$. Suppose that $(q_0, u)\delta = q$. When $\mathfrak{A}_i$ is in state q and reads v, it completes a loop before finishing reading v. (By the discussion above, the loop is simple and uniquely determined.) So v factorizes as $v'v''v'''$, with $|v''| > 0$, such that $(q, v')\delta = (q, v'v'')\delta = q'$. Assume that $|v'|$ is minimal, so that q' is the first state on the loop that $\mathfrak{A}_i$ encounters while reading v.

Assume further that $|v''|$ is minimal, so that $\mathfrak{A}_i$ makes exactly one circuit around the loop while reading v''. Now, by definition, D is a multiple of $|v''|$. Let $m = D/|v''|$. So $|v(v'')^{m+1}v'''| = |v| + D$. By the pumping lemma, $uv'(v'')^{m+1}v'''w \in \operatorname{conv}(L(\mathfrak{A}_i))$.

Consider what this means in terms of the tuple $\vec{p} = (a^{p_1}, \ldots, a^{p_{r_i}})$ such that $\operatorname{conv}(\vec{p}) = uvw$. Since $v' \in b^*$, it follows that

$$uv'(v'')^{m+1}v'''w = \operatorname{conv}(a^{p_1+q_1}, \ldots, a^{p_{r_i}+q_{r_i}}),$$

where

$$q_j = \begin{cases} 0 & \text{if } p_j \leq |u| \\ D & \text{if } p_j \geq |uv|. \end{cases}$$

(Note that either $p_j \leq |u|$ or $p_j \geq |uv|$ since $v \in b^*$ for a fixed $b \in \{a, \$\}^{r_i}$.) Therefore we have the following:

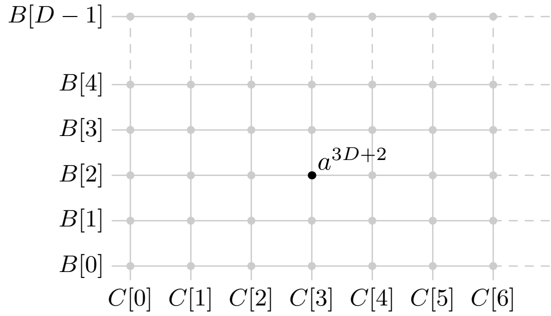
Pumping rule 1. If the components of a tuple in $\Lambda(R_i, \phi)$ can be partitioned into those that are of length at most $l \in \mathbb{N}$ and those that have length at least $l + D$, then [the word encoding] this tuple can be pumped so as to increase by D the lengths of those components that are at least $l + D$ letters long and yield another [word encoding a] tuple in $\Lambda(R_i, \phi)$.

(Notice that this also applies when all components have length at least D; in this case, set $l = 0$.)

With the same setup as above, suppose $|v| \geq 2D$. Then $\mathfrak{A}_i$ must follow the loop labelled by v'' starting at q' at least $m = D/|v''|$ times. That is, v factorizes as $v'(v'')^m\tilde{v}'''$. By the pumping lemma, $uv'\tilde{v}''' \in \operatorname{conv}(L(\mathfrak{A}_i))$ and $|v'\tilde{v}'''| = |v| - D$. Therefore, we also have the following:

Pumping rule 2. If the components of a tuple in $\Lambda(R_i, \phi)$ can be divided into those that are of length less than $l \in \mathbb{N}$ and those that have length at least $l + 2D$, then [the word encoding] this tuple can be pumped so as to decrease by D the length of those components that are at least $l + 2D$ letters long and yield another [word encoding a] tuple in $\Lambda(R_i, \phi)$.

This ability to pump so as to increase or decrease lengths of components by a constant D lends itself to a very useful diagrammatic representation of the unary FA-presentation (a^*, ϕ). Consider a grid of D rows and infinitely many columns. The rows, from bottom to top, are $B[0], \ldots, B[D-1]$. The columns, starting from the left, are $C[0], C[1], \ldots$. The point in column $C[x]$ and row $B[y]$ corresponds to the word a^{xD+y}. For example, in the following diagram, the distinguished point is in column $C[3]$ and row $B[2]$ and so corresponds to a^{3D+2}:



The power of such diagrams is due to a natural correspondence between pumping as in Pumping rules 1 and 2 and certain simple manipulations of tuples of points in the diagram. Before describing this correspondence, we must set up some notation. We will not distinguish between a point in the grid and the word to which it corresponds. The columns are ordered in the obvious way, with $C[x] < C[x']$ if and only if $x < x'$. Extend the notation for intervals on $\mathbb{N}$ to intervals of contiguous columns. For example, for $x, x' \in \mathbb{N}$ with $x \leq x'$, let $C[x, x')$ denotes the set of elements in columns $C[x], \ldots, C[x'-1]$, and $C(x, \infty)$ denotes the set of elements in columns $C[x+1], C[x+2], \ldots$. For any element $u \in a^*$, let $c(u)$ be the index of the column containing u.

Consider the components of an r_i-tuple $\vec{p}$ in $\Lambda(R_i, \phi)$, viewed as an r_i-tuple of points in the diagram. If there is a column $C[x]$ that contains none of the components of $\vec{p}$, then all the components that lie in $C[0, x)$ are at least D shorter than those lying in $C(x, \infty)$. Hence the word encoding the tuple $\vec{p}$ can be pumped between these two sets of components in accordance with Pumping rule 1. This corresponds to shifting all those components that lie in $C(x, \infty)$ rightwards by one column. The tuple that results after this rightward shift of some components also lies in $\Lambda(R_i, \phi)$. [Notice in particular that if column $C[0]$ contains none of the components of $\vec{p}$, then every component can be shifted right by one column, giving a new tuple that also lies in $\Lambda(R_i, \phi)$.] This rightward shifting of components can be iterated arbitrarily many times to yield new tuples. Thus we have the following diagrammatic version of Pumping rule 1:

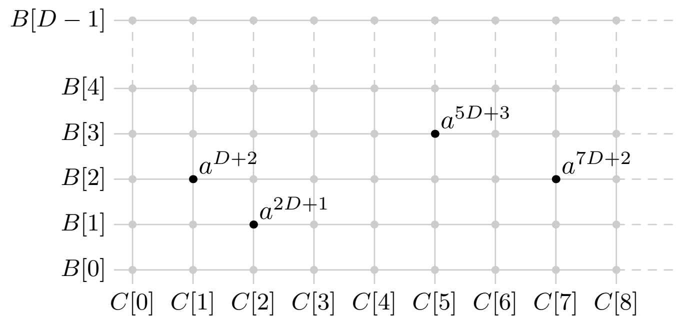
Shift rule 1. Consider the components of an r_i-tuple $\vec{p}$ in $\Lambda(R_i, \phi)$, viewed as an r_i-tuple of points in the diagram. If there is a column $C[x]$ that contains none of the components of $\vec{p}$, then for any $k \in \mathbb{N}$, shifting the components in $C(x, \infty)$ to the right by k columns yields a tuple that also lies in $\Lambda(R_i, \phi)$.

Similarly, if there are two adjacent columns $C[x]$ and $C[x+1]$ that contain none of the components of $\vec{p}$, then every component in $C[0, x)$ is at least $2D$ shorter than every component in $C(x+1, \infty)$. Therefore the word encoding this tuple can be pumped between these sets of components in accordance with Pumping rule 2. This corresponds to shifting all components in $C(x+1, \infty)$ leftwards by one column. The tuple that results after this leftward shift of some components also lies in $\Lambda(R_i, \phi)$. [Notice in particular that if columns $C[0]$ and $C[1]$ contain none of the components of $\vec{p}$, then every component can be shifted left by one column, giving a new tuple that also lies in $\Lambda(R_i, \phi)$.] This leftward shifting of components can be iterated to yield new tuples for as long as the two columns $C[x]$ and $C[x+1]$ do not contain any elements of the latest tuple. Thus we have the following diagrammatic version of Pumping rule 2:

Shift rule 2. Consider the components of an r_i-tuple $\vec{p}$ in $\Lambda(R_i, \phi)$, viewed as an r_i-tuple of points in the diagram. If the columns in $C[x, x+h]$ contain none of the components of $\vec{p}$, then for any k with $0 < k \leq h$, shifting the components in $C(x+h, \infty)$ to the left by k columns yields a tuple that also lies in $\Lambda(R_i, \phi)$.

For convenience, define for every $n \in \mathbb{Z}$ a partial map $\tau_n : a^* \to a^*$, where $a^k \tau_n$ is defined to be a^{k+nD} if $k + nD \geq 0$ and is otherwise undefined. Notice that if $n \geq 0$, the map τ_n is defined everywhere. In terms of the diagram, $a^k \tau_n$ is the element obtained by shifting a^k to the right by n columns if $n \geq 0$ and to the left by $-n$ columns if $n < 0$. The values of k and $n < 0$ for which $a^k \tau_n$ are undefined are precisely those where shifting a^k to the left by $-n$ columns would carry it beyond the left-hand edge of the diagram.

Example 4.1. In order to illustrate Shift rules 1 and 2, consider a 4-tuple $\vec{p} = (a^{2D+1}, a^{D+2}, a^{5D+3}, a^{7D+2})$. This corresponds to the following four points in the diagram.



Shift rule 1 (or the corresponding Pumping rule 1) can be applied in exactly three ways here:

(1) The column $C[0]$ contains no components of $\vec{p}$, so, by Shift rule 1, for any $k \in \mathbb{N}$, all components can be shifted rightward by k columns, yielding the tuple $(a^{(2+k)D+1}, a^{(1+k)D+2}, a^{(5+k)D+3}, a^{(7+k)D+2})$.

(2) The columns $C[3]$ and $C[4]$ contain no components of $\vec{p}$, so, by Shift rule 1, for any $k \in \mathbb{N}$, the third and fourth components can be shifted rightwards by k columns, yielding the tuple $(a^{2D+1}, a^{D+2}, a^{(5+k)D+3}, a^{(7+k)D+2})$.

(3) The column $C[6]$ contains no components of $\vec{p}$, so, by Shift rule 1, for any $k \in \mathbb{N}$, the fourth component of $\vec{p}$ can be shifted rightwards by k columns, yielding the tuple $(a^{2D+1}, a^{D+2}, a^{5D+3}, a^{(7+k)D+2})$.

Shift rule 2 (or the corresponding Pumping rule 2) can be applied in only one way here: columns $C[3]$ and $C[4]$ contain no component of $\vec{p}$, so the third and fourth components can be shifted leftwards by one column, yielding the tuple $(a^{2D+1}, a^{D+2}, a^{4D+3}, a^{6D+2})$.

5. Unary FA-presentable algebras

This section studies finitely generated subalgebras of unary FA-presentable algebras. The key result, Theorem 5.1, shows that the language representing elements of such a subalgebra is regular and that there is an algorithm that effectively constructs this language. From this it follows that the class of unary FA-presentable algebras is closed under taking finitely generated subalgebras (Theorem 5.2) and that the membership problem for finitely generated subalgebras is decidable (Theorem 5.3). Note that Theorem 5.2 does not hold without the hypothesis of finite generation: an arbitrary subsemigroup of a unary FA-presentable semigroup may not even be FA-presentable [4, Example 9.4].

Theorem 5.1. *Let $\mathcal{S}$ be an algebra that admits a unary FA-presentation (a^*, ϕ) and let $\mathcal{T}$ be a finitely generated subalgebra of $\mathcal{S}$. Let L be the sublanguage of a^* consisting of representatives of elements of $\mathcal{T}$. Then L is regular, and an automaton recognizing L can be constructed effectively from a finite set of words representing a generating set for $\mathcal{T}$.*

We will prove this result using the diagrammatic representation discussed in the previous section. Remark 5.4 outlines an alternative proof technique using monadic second-order logic, pointed out to us by the anonymous referee, and dicusses our reasons for preferring the diagrammatic proof.

Proof. We will first of all show that the language L is regular and then show how an automaton recognizing L can be constructed effectively.

Regularity. Let r be the maximum arity of any of the operations in the signature of $\mathcal{S}$. (That is, r is the maximum of their arities *qua* operations, not *qua* relations.) Choose the constant D in accordance with § 4.

Before embarking on the proof proper, let us describe the strategy informally. For $x, y \in \mathbb{N}^0$ with $x \leq y$, define new notation $L[x] = C[x] \cap L$ and $L[x, y] = C[x, y] \cap L$. So $L[x]$ is the set of points in the column $C[x]$ representing elements of the subalgebra. Our strategy will be to prove that the sequence of sets $L[x]$ becomes 'periodic', in the sense that for some $h, h' \in \mathbb{N}^0$ with $h < h'$, the column $L[h']$ is identical to $L[h]$ shifted to the right by $h' - h$ columns. This 'periodicity' is sufficient to prove L is regular.

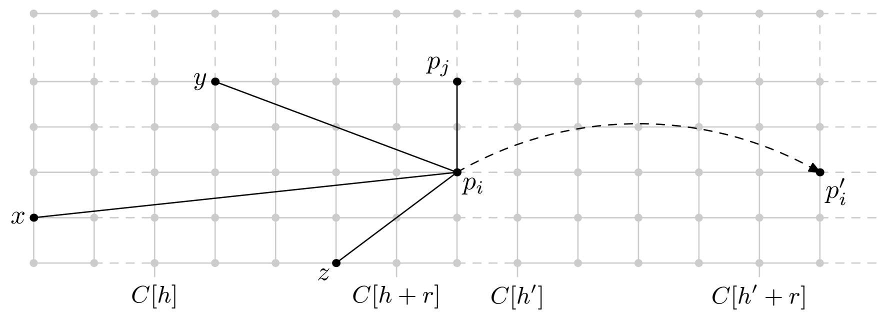


FIGURE 3. Here, the indices h and h' are such that $L[h, h+r]\tau_{h'-h} \subseteq L[h', h'+r]$. The point p'_i is defined to be $p_i\tau_{h'-h}$. The solid lines indicate how $p_i\phi$ is obtained by an application of some operation to the elements $x\phi$, $y\phi$, $z\phi$, and $p_j\phi$ (where $j < i$).

To prove this 'periodicity', we will show that for some fixed m, and any $h' > h > m$ the set $L[h+r+1]$ is 'almost' determined by the set $L[h, h+r]$ and the set $L[0, m]$, in the following sense: If every element in $L[h, h+r]$, when shifted to the right by $h'-h$ columns, gives an element $L[h', h'+r]$ (condition (5.1) below), then every element in $L[h+r+1]$, when shifted to the right by $h'-h$ columns, gives an element $L[h'+r+1]$ (condition (5.2) below). Now, 'new' elements that do not correspond to $L[h+r+1]$ may appear in $L[h'+r+1]$, but because each column contains only D elements, 'new' elements can appear only finitely many times, after which our 'periodicity' must hold.

The rest of the proof of the regularity of L is simply a formalization of this strategy.

First, since the subalgebra $\mathcal{T}$ is finitely generated, it is generated by the elements in $L[0, m]\phi$ for some $m \in \mathbb{N}^0$.

Suppose that $h, h' \in \mathbb{N}^0$, where $h' > h > m$, are such that

$$L[h, h+r]\tau_{h'-h} \subseteq L[h', h'+r]. \tag{5.1}$$

The immediate aim is to prove that

$$L[h+r+1]\tau_{h'-h} \subseteq L[h'+r+1]. \tag{5.2}$$

These conditions are formal restatements of those described above.

Because $h+r > m$, the elements of $L[0, h+r]\phi$ generate the elements of $L[h+r+1]\phi$. That is, by applying the operations of $\mathcal{S}$ to elements of $L[0, h+r]\phi$, one can obtain a finite sequence of points $p_1, \ldots, p_n \in L[h+r+1, \infty)$ such that each $p_i\phi$ is obtained by a single application of some operation to elements from $(L[0, h+r] \cup \{p_1, \ldots, p_{i-1}\})\phi$, and such that $L[h+r+1]\phi \subseteq \{p_1, \ldots, p_n\}\phi$. [It may be necessary for some p_i to lie in columns to the right of $C[h+r+1]$, in order to later generate the elements of $L[h+r+1]\phi$.]

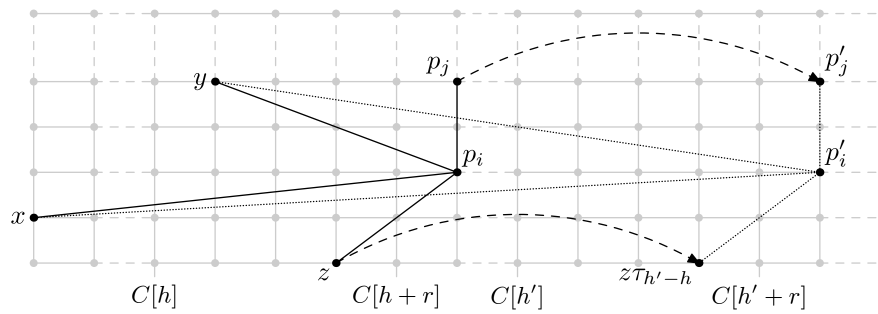


FIGURE 4. The dotted lines indicate how p'_i is obtained by an application of the same operation to the elements x, y, $z\tau_{h'-h}$, and $p'_j = p_j\tau_{h'-h}$.

For each $i \in \{1,\dots,n\}$, let $p'_i = p_i\tau_{h'-h}$. The aim is to prove by induction on i that $p'_i \in L$. We will show that, just as $p_i\phi$ is obtained by an application of some operation to elements from $(L[0,h+r] \cup \{p_1,\dots,p_{i-1}\})\phi$, so $p'_i\phi$ can be obtained by an application of the same operation to elements of

$$\big(L[0,h+r] \cup L[h',h'+r] \cup \{p'_1,\dots,p'_{i-1}\}\tau_{h'-h}\big)\phi.$$

So suppose that $p'_1,\dots,p'_{i-1} \in L$. Suppose $p_i\phi = (x_1\phi,\dots,x_k\phi)f$, where f is an operation of arity $k \leq r$ and $x_1,\dots,x_k \in L[0,h+r]\cup\{p_1,\dots,p_{i-1}\}$. (See Figure 3.) Without loss of generality, assume that $c(x_j) \leq c(x_{j+1})$ for all $j \in \{1,\dots,k-1\}$. Since $k \leq r$, there is at least one column C' in $C[h],\dots,C[h+r]$ that does not contain any point $x_1,\dots,x_k$. Let $x_1,\dots,x_j$ be the points lying to the left of this column, and $x_{j+1},\dots,x_k$ be those lying to the right.

For $l \in \{j+1,\dots,k\}$, let $x'_l = x_l\tau_{h'-h}$. Recall that $p'_i = p_i\tau_{h'-h}$. Now, since $x_l \in L[h,h+r]$, it follows that $x'_l \in L[h',h'+r]$ by (5.1). On the other hand, if x'_l is one of the points $p'_1,\dots,p'_{i-1}$, then it lies in L by the induction hypothesis. The application of the operation f to the elements $x_1\phi,\dots,x_k\phi$ gives $p_i\phi$. Let $\vec{p} = (x_1,\dots,x_k,p_i) \in \Lambda(f,\phi)$. Then by Shift rule 1, the tuple

$$\vec{q} = (x_1,\dots,x_j,x'_{j+1},\dots,x'_k,p'_i),$$

obtained by shifting rightwards the components $x_{j+1},\dots,x_k,p_i$, also lies in $\Lambda(f,\phi)$. Since all of $x_1\phi,\dots,x_j\phi$ and $x'_{j+1}\phi,\dots,x'_k\phi$ lie in the subalgebra $\mathcal{T}$, so does $p'_i\phi$. (See Figure 4.) Hence $p'_i \in L$.

Therefore, by induction, all the points p'_i lie in L, and hence condition (5.2) holds. Thus condition (5.1) entails condition (5.2).

Since each of the sets $L[h,h+r]$ contains at most $(r+1)D$ elements, there must exist $h,h' \in \mathbb{N}^0$ with $h' > h$ such that (5.1) holds. Fix two such values h and h'. Then it follows by induction on i that $L[i]\tau_{h'-h} \subseteq L[i+h'-h]$ for all $i \geq h$. Since the size of the sets $L[i]$ is bounded above by D, there exists $g \in \mathbb{N}$ such that $L[i]\tau_{h'-h} = L[i+h'-h]$ for all $i \geq g$. Thus

$$L = L[0,g-1] \cup (a^{D(h'-h)})^* L[g,g+h'-h-1]$$

and so is regular.

Effective construction. Let L_0 be a finite set of words representing a generating set for the subalgebra $\mathcal{T}$. We will inductively construct a sequence of regular sublanguages L_i of L for $i \in \mathbb{N}^0$. From some point onwards, every language in this sequence will be L itself. We will be able to detect when $L_i = L$, but we cannot bound in advance the number of terms we must compute before obtaining L. For all $i, x, y \in \mathbb{N}^0$ with $x \leq y$, let $L_i[x] = C[x] \cap L_i$ and $L_i[x, y] = C[x, y] \cap L_i$.

Inductively define the language L_{i+1} as follows: find the minimal h such that there exists h' such that

$$L_i[h, h + r]\tau_{h'-h} \subseteq L_i[h', h' + r].$$

(Notice that this is (5.1) restated with L_i in place of L.) Let h_i be h and let h'_i be minimal among corresponding such h', and let

$$L_{i+1} = L_i \cup \big((a^{D(h'_i - h_i)})^* L_i[h_i, h'_i - 1]\big) \cup K_{i+1}$$

where

$$K_{i+1} = \big\{(s_1, \ldots, s_{r_j}, x)f_j : \\ j \in \{1, \ldots, k\}, s_1, \ldots, s_{r_j} \in L_i\phi, f_j \text{ has arity } r_j\big\}\phi^{-1}.$$

[Notice that h_i and h'_i always exist since because the sets $L_i[h_i, h'_i + r]$ are all of bounded size. Notice further that when L_i is finite, $L_i[h_i, h'_i + r]$ may be empty. Observe that h_i and h'_i can be found simply by enumerating sets $L_i[h, h + r]$.]

Let us prove by induction that $L_i \subseteq L$. Clearly $L_0 \subseteq L$. Suppose that $L_i \subseteq L$. By the reasoning in used in the proof of regularity above, each element of $(a^{D(h'_i - h_i)})^* L_i[h_i, h'_i - 1]$ lies in L. The language K_{i+1} consists of representatives of elements obtained by applying the operations of $\mathcal{S}$ to elements of $L_i\phi$. Since $\mathcal{T}$ is a subalgebra, every element of the language K_{i+1} thus lies in L. Hence $L_{i+1} \subseteq L$.

Furthermore, the language K_{i+1} consists of representatives of elements satisfying a first-order formula. Hence, if L_i is regular and given by a finite automaton, a finite automaton recognizing L_{i+1} can be effectively constructed. Since L_0 is finite, it follows by induction that every L_i is regular, and that for any $i \in \mathbb{N}^0$ an automaton recognizing L_i can be effectively constructed.

Notice further that $L_i \subseteq L_{i+1}$ and that for any $u \in L$, there exists some L_i such that $u \in L_i$.

By the reasoning in the proof of regularity above, there exist $g, h, h' \in \mathbb{N}$ be such that $L[i]\tau_{h'-h} = L[i + h' - h]$ for all $i \geq g$. Note that $L = L[0, g - 1] \cup (a^{D(h'-h)})^* L[g, g + h' - h]$. Let n be such that $L[0, g + h' - h] \subseteq L_n$. Then, by definition, L_{n+1} contains L_n and $(a^{D(h'-h)})^* L[g, g + h' - h]$. Hence $L \subseteq L_{n+1}$.

Therefore, the algorithm constructing the various L_i will at some point construct $L_{n+1} = L$. Furthermore, the algorithm can check whether L_i is L

simply by checking whether $L_i = L_{i+1}$, for if this holds, then $L_i\phi$ is closed under all the operations of $\mathcal{S}$ and hence must be the domain of the subalgebra $\mathcal{T}$. Thus there is an effective procedure that constructs L. $\square$

Theorem 5.2. *The class of unary FA-presentable algebras is closed under taking finitely presented subalgebras.*

Proof. Let $\mathcal{S}$ be an algebra that admits a unary FA-presentation (a^*, ϕ) and let $\mathcal{T}$ be a finitely generated subalgebra of $\mathcal{S}$. Let L be the sublanguage of a^* consisting of representatives of elements of $\mathcal{T}$. By Theorem 5.1, L is regular, whence

$$\Lambda(R, \phi|_L) = \Lambda(R, \phi) \cap \big(\underbrace{L \times L \times \ldots \times L}_{k \text{ times}}\big)$$

for any k-ary relation (or operation) R of $\mathcal{S}$, which shows that $(L, \phi|_L)$ is a unary FA-presentation for $\mathcal{T}$. $\square$

The following theorem deals with the membership problem for finitely generated subalgebras of unary FA-presentable algebras. This problem is not decidable for general FA-presentable algebras, because reachability in the configuration graph of a Turing machine is undecidable.

Theorem 5.3. *There is an algorithm that takes a unary FA-presentation (a^*, ϕ) for an algebra $\mathcal{S}$, a finite set X of words in a^*, and a word $w \in a^*$, and decides whether $w\phi$ lies in the subalgebra generated by $X\phi$.*

Proof. By Theorem 5.1, there is an algorithm that takes the finite set of words X and constructs the sublanguage L of a^* consisting of representatives of elements of the subalgebra $\mathcal{T}$ generated by $X\phi$. To decide whether $w\phi$ lies in $\mathcal{T}$, it remains to check whether w lies in L. $\square$

Remark 5.4. The anonymous referee also pointed out that Theorem 5.1 and Theorem 5.2 could also be proven by using the fact that unary FA-presentable structures are monadic second-order definable in $(\mathbb{N}, <)$. The idea is as follows: membership of the subset of $\mathbb{N}$ corresponding to the subalgebra can be defined by a monadic second-order formula that says (informally) that x belongs to the smallest (with respect to inclusion) subset of $\mathbb{N}$ that contains all the generators and is closed with respect to the operations of the algebra. This formula can then be effectively translated back to an automaton recognizing the language words a^n such that n satisfies this formula (see, for example, [16]).

However, the diagrammatic proof above has the advantage of using purely automata-theoretic methods, without having to translate to monadic second-order logic and back. Although passing to and from the monadic second-order formula is effective, the underlying constructions are intricate (see [16, Theorem 3.1]), and in this context they obscure how the unary automatic presentation governs the subalgebras and interacts with the language of representatives for the subalgebra. The diagrammatic proof is much more transparent in this regard. Moreover, the diagrammatic proof has certain parallels with the authors' proof of that the class of unary FA-presentable binary relations is closed under taking transitive closures [3, Theorem 4.1], and serves to illustrate the breadth of results that can be proved using diagrams.

Let $\mathcal{S}$ be a finitely generated algebra. Let $f_1, \ldots, f_k$ be the operations of $\mathcal{S}$. Let G_0 be a finite generating set for $\mathcal{S}$. Inductively define the following finite sets for all $i \in \mathbb{N}$:

$$G_i = G_{i-1} \cup \big\{(s_1, \ldots, s_{r_j})f_j : \\ j \in \{1, \ldots, k\}, s_1, \ldots, s_{r_j} \in G_{i-1}, f_j \text{ has arity } r_j\big\}.$$

Define $g : \mathbb{N}^0 \to \mathbb{N}^0$ by $n \mapsto |G_n|$. The function g is called the *growth level* of $\mathcal{S}$ with respect to the generating set G_0. (This definition is taken from [8, § 4].) If $\mathcal{S}$ is FA-presentable, then there exist constants $s, a, b \in \mathbb{N}$ such that $(n)g \leq s^{a+1+bn}$ for all $n \in \mathbb{N}^0$ [8, Lemma 4.5].

Proposition 5.5. *If $\mathcal{S}$ is a unary FA-presentable algebra, then there exist constants $a, b \in \mathbb{N}$ such that $(n)g \leq a + bn$ for all $n \in \mathbb{N}^0$.*

Proof. Let (a^*, ϕ) be a unary FA-presentation for $\mathcal{A}$. Let $x \in \mathbb{N}^0$ be such that $G_0 \subseteq C[0, x]\phi$. The first aim is to prove, by induction, that $G_n \subseteq C[0, x+n]\phi$ for all $n \in \mathbb{N}^0$. This clearly holds for $n = 0$.

Let f_j be an operation of $\mathcal{S}$ whose arity is r_j. Let $u_1, \ldots, u_{r_j} \in G_i\phi^{-1} \subseteq C[0, x+n]$. Let $v \in a^*$ be such that $(u_1\phi, \ldots, u_{r_j}\phi)f_j = v\phi$. Suppose for *reductio ad absurdum* that $c(v) > x + n + 1$. (Recall that $c(v)$ is the index of the column containing v.) Then no component of the tuple $(u_1, \ldots, u_{r_j}, v) \in \Lambda(f_j, \phi)$ lies in $C[x + n + 1]$ and v is the only component lying in $C[x + n + 2, \infty)$ and so, by Shift rule 1, $(u_1, \ldots, u_{r_j}, v\tau_1) \in \Lambda(f_j, \phi)$. Hence $v\phi = (u_1\phi, \ldots, u_{r_j}\phi)f_j = (v\tau_1)\phi$, which contradicts the injectivity of ϕ. Therefore $v \in C[0, x + n + 1]$. Since $v\phi$ is the result of applying an arbitrary operation f_j of $\mathcal{S}$ to arbitrary elements of G_n, it follows that $G_{n+1} \subseteq C[0, x + n + 1]\phi$.

Let $a = (x + 1)D$; then $|C[0, x]| = a$. Furthermore, $|C[0, x + n + 1]| = |C[0, x + n]| + D$. Let $b = D$; then $|C[0, x + n + 1]| = a + bn$. Since $G_n \subseteq C[0, x + n + 1]\phi$, it follows that $(n)g = |G_n| \leq a + bn$. ☐

The contrast between the growth levels of finitely generated FA-presentable algebras (bounded by an exponential function) and finitely generated *unary* FA-presentable algebras (bounded by a linear function) resembles the contrast between the growth of finitely generated FA-presentable semigroups (polynomial growth [2, Theorem 7.4]) and finitely generated unary FA-presentable semigroups (sublinear growth, which implies finiteness [4, Proof of Theorem 13]). Note, however, the difference between the two types of growth: the growth level of an algebra counts elements of a given term complexity, and the growth of a semigroup or group counts elements of given word length.

References

[1] Blumensath, A.: Automatic structures. Diploma thesis, RWTH Aachen (1999). URL www.mathematik.tu-darmstadt.de/~blumensath/Publications/AutStr.pdf

[2] Cain, A.J., Oliver, G., Ruškuc, N., Thomas, R.M.: Automatic presentations for semigroups. Inform. and Comput. **207**(11), 1156–1168 (2009). DOI 10.1016/j.ic.2009.02.005

[3] Cain, A.J., Ruškuc, N.: Unary FA-presentable binary relations: transitivity and classification results. Preprint, arXiv: 1303.0214.

[4] Cain, A.J., Ruškuc, N., Thomas, R.M.: Unary FA-presentable semigroups. Internat. J. Algebra Comput. **22**(4) (2012). DOI 10.1142/S0218196712500385

[5] Delhommé, C.: Automaticité des ordinaux et des graphes homogènes. C. R. Math. Acad. Sci. Paris **339**(1), 5–10 (2004). DOI 10.1016/j.crma.2004.03.035

[6] Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation. Addison–Wesley Publishing Co., Reading, Mass. (1979)

[7] Khoussainov, B., Liu, J., Minnes, M.: Unary automatic graphs: an algorithmic perspective. Math. Structures Comput. Sci. **19**(1), 133–152 (2009). DOI 10.1017/S0960129508007342. URL http://dx.doi.org/10.1017/S0960129508007342

[8] Khoussainov, B., Nerode, A.: Automatic presentations of structures. In: Logic and computational complexity (Indianapolis, IN, 1994), *Lecture Notes in Computer Science*, vol. 960, pp. 367–392. Springer, Berlin (1995). DOI 10.1007/3-540-60178-3_93

[9] Khoussainov, B., Rubin, S., Stephan, F.: Automatic partial orders. In: Proceedings of the 18th IEEE Symposium on Logic in Computer Science, pp. 168–177. IEEE Computer Society (2003). DOI 10.1109/LICS.2003.1210056

[10] Khoussainov, B., Rubin, S., Stephan, F.: Automatic linear orders and trees. ACM Trans. Comput. Log. **6**(4), 675–700 (2005). DOI 10.1145/1094622.1094625

[11] Liu, J., Minnes, M.: Deciding the isomorphism problem in classes of unary automatic structures. Theoret. Comput. Sci. **412**(18), 1705–1717 (2011). DOI 10.1016/j.tcs.2010.12.045. URL http://dx.doi.org/10.1016/j.tcs.2010.12.045

[12] Nies, A., Semukhin, P.: Finite automata presentable abelian groups. Ann. Pure Appl. Logic **161**(3), 458–467 (2009). DOI 10.1016/j.apal.2009.07.006. URL http://dx.doi.org/10.1016/j.apal.2009.07.006

[13] Nies, A., Thomas, R.M.: FA-presentable groups and rings. J. Algebra **320**(2), 569–585 (2008). DOI 10.1016/j.jalgebra.2007.04.015. URL http://dx.doi.org/10.1016/j.jalgebra.2007.04.015

[14] Oliver, G.P., Thomas, R.M.: Automatic presentations for finitely generated groups. In: V. Diekert, B. Durand (eds.) 22nd Annual Symposium on Theoretical Aspects of Computer Science (STACS'05), Stuttgart, Germany, *Lecture Notes in Comput. Sci.*, vol. 3404, pp. 693–704. Springer, Berlin (2005). DOI 10.1007/978-3-540-31856-9_57

[15] Rubin, S.: Automatic structures. Ph.D. Thesis, Auckland University (2005)

[16] Thomas, W.: Languages, automata, and logic. In: Handbook of formal languages, Vol. 3, pp. 389–455. Springer, Berlin (1997)

Alan J. Cain

Centro de Matemática, Faculdade de Ciências, Universidade do Porto, Rua do Campo Alegre 687, 4169–007 Porto, Portugal
e-mail: ajcain@fc.up.pt
URL: http://www.fc.up.pt/pessoas/ajcain

Nik Ruškuc

School of Mathematics and Statistics, University of St Andrews, North Haugh, St Andrews, Fife KY16 9SS, United Kingdom
e-mail: nik@mcs.st-andrews.ac.uk
URL: http://turnbull.mcs.st-andrews.ac.uk/~nik