RAIRO-Theor. Inf. Appl. 46 (2012) 573–592
DOI: 10.1051/ita/2012014

Available online at:
www.rairo-ita.org

# ON BIAUTOMATA *

Ondřej Klíma[1] and Libor Polák[1]

**Abstract.** We initiate the theory and applications of biautomata. A biautomaton can read a word alternately from the left and from the right. We assign to each regular language $L$ its canonical biautomaton. This structure plays, among all biautomata recognizing the language $L$, the same role as the minimal deterministic automaton has among all deterministic automata recognizing the language $L$. We expect that from the graph structure of this automaton one could decide the membership of a given language for certain significant classes of languages. We present the first two results of this kind: namely, a language $L$ is piecewise testable if and only if the canonical biautomaton of $L$ is acyclic. From this result Simon's famous characterization of piecewise testable languages easily follows. The second class of languages characterizable by the graph structure of their biautomata are prefix-suffix testable languages.

**Mathematics Subject Classification.** 68Q70.

## 1. Introduction

Regular languages are recognized, among others, by deterministic automata. A regular language $L$ possesses, up to isomorphism, a unique minimal complete deterministic automaton. There is a construction due to Brzozowski [1] where the states are constructed as left derivatives (sometimes also called left quotients) of $L$. A useful property of this canonical automaton is that each state $q$ is a language

*Keywords and phrases.* Biautomata, canonical biautomaton, piecewise testable languages, prefix-suffix languages.

* *Both authors were supported by the Ministry of Education of the Czech Republic under the project MSM 0021622409 and by the Grant 201/09/1313 of the Grant Agency of the Czech Republic.*

[1] Department of Mathematics and Statistics, Masaryk University, Kotlářská 2, 61137 Brno, Czech Republic. polak@math.muni.cz

Article published by EDP Sciences

© EDP Sciences 2012

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

and it is exactly the set of all words transforming $q$ into a terminal state. A similar view concerning the states was also applied in the theory of universal automata, see Lombardy and Sakarovitch [5] and Polák [8]. Namely, the states of the universal automaton are exactly the finite intersections of left derivatives. This led the authors to consider the so-called meet automata [3]. In this setting the universal automaton of a language $L$ can be viewed as the canonical meet automaton for $L$. The algebraic approach uses other structures for a language $L$, namely the syntactic monoid and the syntactic semiring of $L$. It is well-known that the syntactic monoid is isomorphic to the transformation monoid of the minimal automaton of $L$. Similarly, the syntactic semiring of $L$ is isomorphic to the transformation semiring of the canonical meet automaton.

One of the major goals in regular language theory is to determine whether a given language is a member of certain significant classes of languages. All the above mentioned structures turned out to be appropriate tools for answering such kind of problems. In this paper we introduce a new structure, called a biautomaton, and we claim that this structure can also clarify some aspects of these questions. Notice that the term "biautomaton" was used by other authors having different meanings. Moreover, our notion is not related to two-way automata.

Motivated by Brzozowski's construction, we consider two-sided derivatives of $L$, to get the states of a new type of automaton. Now each letter determines two independent actions on states, namely the derivative from the left and the derivative from the right. In such a way, we get the so-called canonical biautomaton and a natural generalization leads to an abstract notion of biautomata. The canonical biautomaton of the language $L$ plays, among all biautomata recognizing $L$, the same role as the minimal deterministic automaton has among all deterministic automata recognizing $L$.

As the first application of the theory of biautomata we give an effective characterization of piecewise testable languages *via* their canonical biautomata. The class of piecewise testable languages is a prominent one in the algebraic theory of regular languages. Simon [9,10] showed that a language is piecewise testable if and only if its syntactic monoid is $\mathcal{J}$-trivial. This characterization is based on one of Green's relations, a basic concept of semigroup theory. Similar classes of monoids related to other of Green's relations are classes of $\mathcal{R}$-trivial monoids and $\mathcal{L}$-trivial monoids, two classes which are right-left dual. It is well-known that a finite monoid is $\mathcal{J}$-trivial if and only if it is simultaneously $\mathcal{R}$-trivial and $\mathcal{L}$-trivial. Apart from the combinatorial characterization of regular languages having $\mathcal{R}$-trivial syntactic monoids, it is known that these languages are those which have acyclic minimal automata (see Sect. 4.3 in [6] for more details). From this point of view, a language $L$ is piecewise testable if and only if both the minimal automaton of $L$ and the minimal automaton of $\overline{L}$ (the left-right dual of the language of $L$) are acyclic. Since both these automata can be found in the canonical biautomaton of $L$, this leads us to the claim that the canonical biautomaton of a piecewise testable language is acyclic as well. We show that this is true and that also the opposite implication is valid.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

**Theorem 1.1.** *Let $L \subseteq A^*$ be a regular language. Then $L$ is piecewise testable if and only if the canonical biautomaton of $L$ is acyclic.*

It is possible to complete the previous arguments into a proof of the theorem as a consequence of known results. Instead of such a proof we show in Section 4 an elementary, direct proof of the theorem. On few pages we give a complete proof which is self-contained. This could demonstrate that there is a certain potential for finding further applications of biautomata in the algebraic theory of regular languages.

Another quite significant type of languages are the so-called prefix-suffix testable languages. We obtain here the following characterization. (Notice that a state is *absorbing* if it is a fixed point for all actions).

**Theorem 1.2.** *Let $L \subseteq A^*$ be a regular language and let $\mathcal{C}_L$ be its canonical biautomaton with the set of states $Q$ and the actions $\cdot$ and $\circ$. Then $L$ is prefix-suffix testable if and only if $\mathcal{C}_L$ satisfies the following condition*

$$(\textit{for each } q \in Q,\ u, v \in A^+)\ q \cdot u = q \circ v = q \textit{ implies that } q \textit{ is absorbing.} \qquad (\dagger)$$

After this introductory section we collect necessary definitions and notation in Section 2. The next section is an introduction to the theory of biautomata. Section 4 characterizes piecewise testable languages in terms of their canonical biautomata. We also derive here the original theorem of Simon from our results. The final section is devoted to a characterization of prefix-suffix testable languages.

## 2. Preliminaries

We fix a finite non-empty alphabet $A$ consisting of *letters*. Let $A^*$ be the free monoid over $A$ with the neutral element $\lambda$, *i.e.* $A^*$ is the set of all *words* over $A$ equipped with the operation of concatenation. For $u = a_1 a_2 \dots a_n \in A^*$ where $n$ is a positive integer and $a_1, a_2, \dots, a_n \in A$, we write $\overline{u} = a_n \dots a_2 a_1$, $|u| = n$ and $\mathsf{c}(u) = \{a_1, \dots, a_n\}$, *i.e.* the set of all letters occurring in $u$. Moreover, we put $\overline{\lambda} = \lambda$, $|\lambda| = 0$ and $\mathsf{c}(\lambda) = \emptyset$. Also, for $L \subseteq A^*$, we write $\overline{L} = \{\, \overline{u} \mid u \in L \,\}$ and $L^{\mathsf{c}} = A^* \setminus L$.

A *complete deterministic* finite automaton over the alphabet $A$ is a five-tuple $\mathcal{A} = (Q, A, \cdot, i, T)$ where

- $Q$ is a nonempty set of *states*;
- $\cdot : Q \times A \to Q$, extended to $\cdot : Q \times A^* \to Q$ by $q \cdot \lambda = q$, $q \cdot (ua) = (q \cdot u) \cdot a$, where $q \in Q$, $u \in A^*$, $a \in A$;
- $i \in Q$ is the *initial* state;
- $T \subseteq Q$ is the set of *terminal* states.

The automaton $\mathcal{A}$ *accepts* the word $u \in A^*$ if $i \cdot u \in T$. The *right language* $\mathscr{L}(\mathcal{A}, q)$ of a state $q \in Q$ with respect to the automaton $\mathcal{A}$ is the set $\{\, w \in A^* \mid q \cdot w \in T \,\}$. The *language recognized* by $\mathcal{A}$ is the set $\mathscr{L}(\mathcal{A}) = \mathscr{L}(\mathcal{A}, i)$ of all words accepted by $\mathcal{A}$.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

For a language $L \subseteq A^*$ and $u \in A^*$, we define $u^{-1}L = \{ w \in A^* \mid uw \in L \}$. Moreover, we put $D_L = \{ u^{-1}L \mid u \in A^* \}$. This set is finite for each regular language $L$. Further, let $\mathcal{D}_L = (D_L, A, \cdot, L, T_L)$, where $q \cdot a = a^{-1}q$, for each $q \in D_L$, $a \in A$, and $T_L = \{ q \in D_L \mid \lambda \in q \}$. This automaton is called the *canonical* automaton for $L$ and it is well-known that it is a minimal complete deterministic automaton for $L$ – see [1].

For a language $L \subseteq A^*$, we define the relation $\equiv_L$ on $A^*$ as follows: for $u, v \in A^*$ we have

$$u \equiv_L v \quad \text{if and only if} \quad (\forall p, r \in A^*)\ (\ pur \in L \iff pvr \in L\ ).$$

The relation $\equiv_L$ is a congruence on $A^*$; it is called the *syntactic congruence* of $L$ and the quotient structure $\mathsf{M}(L) = A^*/\equiv_L = \{ [u]_{\equiv_L} \mid u \in A^* \}$ is called the *syntactic monoid* of $L$. Moreover, the monoid $\mathsf{M}(L)$ is finite whenever $L$ is a regular language. The natural mapping $\eta_L : A^* \to \mathsf{M}(L)$ given by $\eta_L(u) = [u]_{\equiv_L}$ is called the *syntactic* homomorphism. The language $L$ is a union of certain classes of the partition $A^*/\equiv_L$. If we denote $F = \eta_L(L)$ the set of these classes, then $L = \{ u \in A^* \mid \eta_L(u) \in F \}$. When $L$ is fixed, we will write simply $M$ and $[u]$ instead of $\mathsf{M}(L)$ and $[u]_{\equiv_L}$.

In a monoid $N$, the elements $a$ and $b$ are *$\mathcal{J}$-related* if $NaN = NbN$. The monoid $N$ is *$\mathcal{J}$-trivial* if for each pair of $\mathcal{J}$-related elements $a, b \in N$, we have $a = b$. We often denote the neutral element of $N$ by 1.

## 3. Biautomata

In this section we initiate a general theory of biautomata. We define this new structure, we introduce the acceptance condition, we consider congruences and quotient biautomata, We present several possible constructions of biautomata for a given language, their minimalization and we equip them with a graph structure.

### 3.1. General definition, congruences, quotient biautomaton, isomorphism

**Definition 3.1.** A *biautomaton* over a finite non-empty alphabet $A$ is a six-tuple $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$ where

- $Q$ is a nonempty set of *states*;
- $\cdot : Q \times A \to Q$, extended to $\cdot : Q \times A^* \to Q$ by $q \cdot \lambda = q$, $q \cdot (ua) = (q \cdot u) \cdot a$, where $q \in Q$, $u \in A^*$, $a \in A$;
- $\circ : Q \times A \to Q$, extended to $\circ : Q \times A^* \to Q$ by $q \circ \lambda = q$, $q \circ (av) = (q \circ v) \circ a$, where $q \in Q$, $v \in A^*$, $a \in A$;
- $i \in Q$ is the *initial* state;
- $T \subseteq Q$ is the set of *terminal* states;
- for each $q \in Q$, $a, b \in A$, we have $(q \cdot a) \circ b = (q \circ b) \cdot a$;
- for each $q \in Q$, $a \in A$, we have $q \cdot a \in T$ if and only if $q \circ a \in T$.

The last two conditions from the definition are generalized below.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

**Lemma 3.2.** *Let $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$ be a biautomaton. Then*

- *for each $q \in Q$, $u, v \in A^*$, we have $(q \cdot u) \circ v = (q \circ v) \cdot u$,*
- *for each $q \in Q$, $u \in A^*$, we have $q \cdot u \in T$ if and only if $q \circ u \in T$.*

*Proof.* We use induction with respect to $|u| + |v|$ to get the first statement. For $|u|, |v| \leq 1$, the statement is clear. Induction step: let $a \in A$, then

$$(q \cdot au) \circ v = ((q \cdot a) \cdot u) \circ v = ((q \cdot a) \circ v) \cdot u = ((q \circ v) \cdot a) \cdot u = (q \circ v) \cdot au\,,$$

$$(q \cdot u) \circ av = ((q \cdot u) \circ v) \circ a = ((q \circ v) \cdot u) \circ a = ((q \circ v) \circ a) \cdot u = (q \circ av) \cdot u.$$

In the proof of the second statement use the first one and induction with respect to $|u|$. □

In contrast to deterministic automata, where we can take a finite non-empty set of states, choose the actions of letters, the initial and the terminal states arbitrarily, the situation for biautomata is more delicate due to the last two conditions from Definition 3.1.

The biautomaton $\mathcal{B}$ *accepts* a given word $u \in A^*$ if $i \cdot u \in T$. This is equivalent to $i \circ u \in T$. In the definition of acceptance we read $u$ from the left-hand side and transform states according to $\cdot$, in the equivalent condition we read $u$ from the right-hand side and transform states according to $\circ$. Moreover, it allows us an *impatient reading* as described below.

**Lemma 3.3.** *Having a biautomaton $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$, $p \in Q$ and $u \in A^+$ dividing $u = u_1 \dots u_k v_k \dots v_1$ arbitrarily, where $u_1, \dots, u_k, v_k, \dots, v_1 \in A^*$, when reading from $p$, the words $u_1$ first, then $v_1$, then $u_2$, and so on, i.e. we move from $p$ to the state*

$$q = ((\dots((((p \cdot u_1) \circ v_1) \cdot u_2) \circ v_2) \dots) \cdot u_k) \circ v_k\,,$$

*then $q \in T$ if and only if $p \cdot u \in T$.*

*Proof.* Using the first part of Lemma 3.2 repeatedly, we get

$$q = (\dots((((\dots((p \cdot u_1) \cdot u_2) \dots) \cdot u_k) \circ v_1) \circ v_2) \dots) \circ v_k = (p \cdot u_1 u_2 \dots u_k) \circ v_k \dots v_2 v_1.$$

Now $q \in T$ if and only if $(p \cdot u_1 u_2 \dots u_k) \cdot v_k \dots v_2 v_1 = p \cdot u \in T$ by the second part of Lemma 3.2. □

Both lemmas will be used in what follows without explicit mentioning.

The *right language* $\mathscr{L}(\mathcal{B}, q)$ of a state $q$ with respect to the biautomaton $\mathcal{B}$ is the set $\{\, w \in A^* \mid q \cdot w \in T \,\}$. The *language recognized* by $\mathcal{B}$ is the set $\mathscr{L}(\mathcal{B}) = \mathscr{L}(\mathcal{B}, i)$. The state $q \in Q$ of the biautomaton $\mathcal{B}$ is *reachable* if there exist $u, v \in A^*$ such that $q = (i \cdot u) \circ v$.

A relation $\sim$ is a *congruence relation* of the biautomaton $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$ if

- $\sim$ is an equivalence relation on the set $Q$;

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

- for each $p, q \in Q$, $a \in A$, the assumption $p \sim q$ implies that both $p \cdot a \sim q \cdot a$ and $p \circ a \sim q \circ a$;
- for each $p \in T$, $q \in Q$, the assumption $p \sim q$ yields $q \in T$.

We define the *quotient* biautomaton $\mathcal{B}/\sim = (Q/\sim, A, \cdot_\sim, \circ_\sim, [i]_\sim, T/\sim)$ where $([q]_\sim) \cdot_\sim a = [q \cdot a]_\sim$ and $([q]_\sim) \circ_\sim a = [q \circ a]_\sim$ (here $[p]_\sim = \{r \in Q \mid r \sim p\}$). This structure is again a biautomaton. Moreover, it recognizes the same language as $\mathcal{B}$ does.

Two biautomata $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$ and $\mathcal{B}' = (Q', A, \cdot', \circ', i', T')$ are *isomorphic* if there exists a bijection $\varphi : Q \to Q'$, called an *isomorphism*, such that

- for each $q \in Q$, $a \in A$, we have that $\varphi(q \cdot a) = \varphi(q) \cdot' a$ and $\varphi(q \circ a) = \varphi(q) \circ' a$;
- $\varphi(i) = i'$;
- for each $q \in Q$, we have that $q \in T$ if and only if $\varphi(q) \in T'$.

Clearly, isomorphic biautomata recognize the same languages.

### 3.2. Constructions of biautomata

The next construction shows how one can naturally convert a deterministic automaton into a biautomaton recognizing the same language.

Given a complete deterministic automaton $\mathcal{A} = (Q, A, \cdot, i, T)$, we define the structure $\mathcal{A}^{\mathsf{B}} = (Q^{\mathsf{B}}, A, \cdot^{\mathsf{B}}, \circ^{\mathsf{B}}, i^{\mathsf{B}}, T^{\mathsf{B}})$, where

- $Q^{\mathsf{B}} = \{\, (q, P) \mid q \in Q,\ P \subseteq Q \,\}$;
- for each $q \in Q$, $P \subseteq Q$, we have $(q, P) \cdot^{\mathsf{B}} a = (q \cdot a, P)$;
- for each $q \in Q$, $P \subseteq Q$, we have $(q, P) \circ^{\mathsf{B}} a = (\, q, \{\, p \in Q \mid p \cdot a \in P \,\}\,)$;
- $i^{\mathsf{B}} = (i, T)$;
- $T^{\mathsf{B}} = \{\, (q, P) \mid q \in Q,\ P \subseteq Q,\ q \in P \,\}$.

**Lemma 3.4.** *For each complete deterministic automaton $\mathcal{A}$, the structure $\mathcal{A}^{\mathsf{B}}$ is a biautomaton recognizing the same language as $\mathcal{A}$ does.*

*Proof.* Let $q \in Q$, $P \subseteq Q$, $u, v \in A^*$. Then

$$(q, P) \cdot^{\mathsf{B}} u = (q \cdot u, P) \quad \text{and} \quad (q, P) \circ^{\mathsf{B}} u = (\, q, \{\, p \in Q \mid p \cdot u \in P \,\}\,).$$

Each of the above states is terminal if and only if $q \cdot u \in P$.

Moreover, $((q, P) \cdot^{\mathsf{B}} u) \circ^{\mathsf{B}} v = (\, q \cdot u, \{\, p \in Q \mid p \cdot v \in P \,\}\,) = ((q, P) \circ^{\mathsf{B}} v) \cdot^{\mathsf{B}} u$ and therefore $\mathcal{A}^{\mathsf{B}}$ is a biautomaton.

Finally,

$$\begin{aligned} \mathscr{L}(\mathcal{A}^{\mathsf{B}}) &= \{\, w \in A^* \mid (i, T) \cdot^{\mathsf{B}} w \in T^{\mathsf{B}} \,\} = \{\, w \in A^* \mid (i \cdot w, T) \in T^{\mathsf{B}} \,\} \\ &= \{\, w \in A^* \mid i \cdot w \in T \,\} = \mathscr{L}(\mathcal{A}). \end{aligned}$$

□

The biautomaton $\mathcal{A}^{\mathsf{B}}$ is called the *reverse biautomaton* of the automaton $\mathcal{A}$.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

The following construction yields another model for a biautomaton accepting a given regular language $L \subseteq A^*$. For $v \in A^*$, we define

$$Lv^{-1} = \{\, w \in A^* \mid wv \in L \,\}, \quad E_L = \{\, Lv^{-1} \mid v \in A^* \,\}, \quad P_L = D_L \times E_L.$$

Now we define $\mathcal{P}_L = (P_L, A, \cdot, \circ, (L, L), T)$, where

$$(s,t)\cdot a = (a^{-1}s, t),\ (s,t)\circ a = (s, ta^{-1}),\ T = \{\, (u^{-1}L, Lv^{-1}) \mid u, v \in A^*, uv \in L \,\}.$$

**Lemma 3.5.** *The above structure $\mathcal{P}_L$ is a biautomaton isomorphic to the biautomaton of all reachable states of $(\mathcal{D}_L)^{\mathsf{B}}$.*

*Proof.* Recall the definition of the automaton $\mathcal{D}_L$ from Section 2. In $\mathcal{P}_L$ we have $((L,L)\cdot u)\circ v = (u^{-1}L, Lv^{-1})$ and in $(\mathcal{D}_L)^{\mathsf{B}}$ we have

$$\begin{aligned}((L,T_L)\cdot u)\circ v &= (\, u^{-1}L, \{\, w^{-1}L \mid w^{-1}L \cdot v \in T_L \,\}\,)\\ &= (\, u^{-1}L, \{\, w^{-1}L \mid (wv)^{-1}L \in T_L \,\})\\ &= (\, u^{-1}L, \{\, w^{-1}L \mid wv \in L \,\}\,)\\ &= (\, u^{-1}L, \{\, w^{-1}L \mid w \in Lv^{-1} \,\}\,).\end{aligned}$$

Let $Q$ be the set of all reachable states in $(\mathcal{D}_L)^{\mathsf{B}}$. Denoting $S_v = \{\, w^{-1}L \mid w \in Lv^{-1} \,\}$ we see that $S_v$ is fully determined by $Lv^{-1}$. Thus the mapping $\alpha : P_L \to Q$ given by $(u^{-1}L, Lv^{-1}) \mapsto (\, u^{-1}L, S_v\,),\ u, v \in A^*$, is correctly defined. We show that $\beta : Q \to P_L$ given by $(\, u^{-1}L, S_v\,) \mapsto (u^{-1}L, Lv^{-1}),\ u, v \in A^*$, is also correctly defined. Indeed, let $v, v', w \in A^*$ be such that $S_v \subseteq S_{v'}$ and $w \in Lv^{-1}$. We will show that $w \in Lv'^{-1}$. Now $w^{-1}L \in S_v$ and hence $w^{-1}L \in S_{v'}$. Therefore there exists $w' \in A^*$ such that $w^{-1}L = w'^{-1}L$ and $w' \in Lv'^{-1}$, *i.e.* $w'v' \in L$. Hence $v' \in w'^{-1}L = w^{-1}L$ and we obtain $wv' \in L$ from which $w \in Lv'^{-1}$ follows. We can interchange $v$ and $v'$ to get the second inclusion in $Lv^{-1} = Lv'^{-1}$. Clearly $\alpha$ and $\beta$ are mutually inverse bijections and $\alpha$ respects the actions of letters. One can check that $\alpha$ maps the initial state onto the initial state and $q \in P_L$ is terminal if and only if $\alpha(q)$ is terminal. □

The biautomaton $\mathcal{P}_L$ is called the *product biautomaton* of the language $L$.

### 3.3. Canonical biautomaton

Now we present a construction analogous to Brzozowski's procedure producing the minimal deterministic automaton of a given regular language. Here we use two-sided derivatives.

For a language $L \subseteq A^*$ and $u, v \in A^*$, we define

$$u^{-1}Lv^{-1} = \{\, w \in A^* \mid uwv \in L \,\}, \quad C_L = \{\, u^{-1}Lv^{-1} \mid u, v \in A^* \,\}.$$

We define $\mathcal{C}_L = (C_L, A, \cdot, \circ, L, T)$, where

$$q \cdot a = a^{-1}q, \quad q \circ a = qa^{-1} \quad \text{and} \quad T = \{\, u^{-1}Lv^{-1} \mid \lambda \in u^{-1}Lv^{-1} \,\}.$$

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

**Lemma 3.6.** *For each regular language $L$ over $A$, the structure $\mathcal{C}_L$ is a biautomaton. Moreover, for each state $q$, the right language $\mathscr{L}(\mathcal{C}_L, q)$ is equal to $q$. In particular, the biautomaton $\mathcal{C}_L$ recognizes the language $L$.*

*Proof.* Let $u, v, w \in A^*$. Realize that each of the states $u^{-1}Lv^{-1}\cdot w$ and $u^{-1}Lv^{-1}\circ w$ is terminal if and only if $uwv \in L$. Since the other defining conditions are trivially satisfied the structure $\mathcal{C}_L$ is a biautomaton. Furthermore, for $q = u^{-1}Lv^{-1}$ we have

$$\mathscr{L}(\mathcal{C}_L, q) = \{\, w \in A^* \mid u^{-1}Lv^{-1} \cdot w \in T \,\} = \{\, w \in A^* \mid uwv \in L \,\} = u^{-1}Lv^{-1}. \ \square$$

The biautomaton $\mathcal{C}_L$ is called the *canonical biautomaton* of the language $L$.

**Example 3.7.** Let $L = \{a, b\}^* ca \{b, c\}^*$ be a language over the alphabet $A = \{a, b, c\}$. In Figure 1, the "right" actions by letters are drawn by dashed arrows. We omit here the empty set state and arrows leading there. The initial state $i$ is the language $L$ and the terminal states are $f_1 = \{a, b\}^*$, $f_2 = \{b, c\}^*$ and $f_3 = \{\lambda\}$. The last two states are $p = \{a, b\}^* c$ and $q = a\{b, c\}^*$. The reader could try to read the word $acab$ from the state $i$ in various ways: $i \cdot acab$, $i \circ acab$ or, for example, $((i \cdot a) \circ ab) \cdot c$.

### 3.4. Minimalization of biautomata

The minimalization procedure for biautomata is similar to that for deterministic automata:

**Lemma 3.8.** *Let $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$ be an arbitrary biautomaton where all states are reachable. Then the relation $\sim$ defined on $Q$ by*

$$p \sim q \quad \text{if and only if} \quad \mathscr{L}(\mathcal{B}, p) = \mathscr{L}(\mathcal{B}, q)$$

*is a congruence relation on $\mathcal{B}$. Moreover, the mapping*

$$\varphi : [(i \cdot u) \circ v]_\sim \ \mapsto u^{-1}Lv^{-1}$$

*is an isomorphism of the quotient biautomaton $\mathcal{B}/\!\sim$ onto the canonical biautomaton for the language $L = \mathscr{L}(\mathcal{B})$.*

*Proof.* Let $L = \mathscr{L}(\mathcal{B})$. An arbitrary state $p \in Q$ is of the form $p = (i \cdot u) \circ v$, $u, v \in A^*$. Then

$$\begin{aligned} \mathscr{L}(\mathcal{B}, (i \cdot u) \circ v) &= \{\, w \in A^* \mid ((i \cdot u) \circ v) \cdot w \in T \,\} \\ &= \{\, w \in A^* \mid i \cdot uwv \in T \,\} \\ &= \{\, w \in A^* \mid uwv \in L \,\} = \ u^{-1}Lv^{-1}. \end{aligned}$$

Thus for $u, v, u', v' \in A^*$, we have

$$p = (i \cdot u) \circ v \sim q = (i \cdot u') \circ v' \ \text{ if and only if } \ u^{-1}Lv^{-1} = (u')^{-1}L(v')^{-1}.$$

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

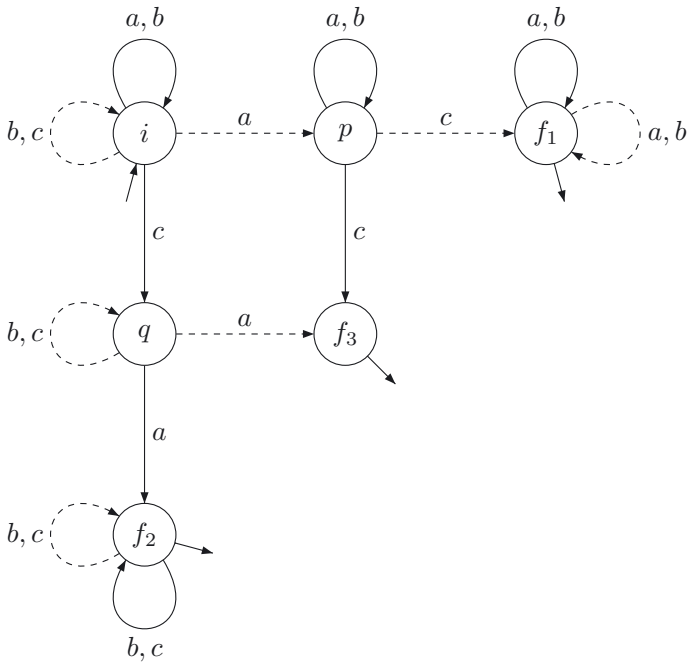



FIGURE 1. The canonical biautomaton of the language $L = \{a, b\}^* ca \{b, c\}^*$.

Now, for each $a \in A$, we have $p \cdot a = ((i \cdot u) \cdot a) \circ v = (i \cdot ua) \circ v$ and $p \circ a = (i \cdot u) \circ (av)$ and similarly for $q$. Thus $p \sim q$ yields both $p \cdot a \sim q \cdot a$ and $p \circ a \sim q \circ a$.

Furthermore, the following statements are equivalent:

$$(i \cdot u) \circ v \in T; \quad i \cdot uv = (i \cdot u) \cdot v \in T; \quad uv \in L; \quad \lambda \in u^{-1} L v^{-1}.$$

Thus $p \in T$, $p \sim q$ implies $q \in T$.

The second part of our statement follows also from the considerations above. □

A biautomaton $\mathcal{B} = (Q, A, \cdot, \circ, i, T)$ is *minimal* for a language $L$ if the congruence $\sim$ from the previous lemma is the diagonal relation on $Q$. We saw that the canonical biautomaton for $L$ is minimal for $L$. Moreover each minimal biautomaton for $L$ is isomorphic to the canonical one.

### 3.5. From syntactic monoid to canonical biautomaton

Here we present another construction of the canonical biautomaton of $L$. We will see its usefulness in applications.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

Keeping the notation from Section 2, notice that also every derivative $u^{-1}Lv^{-1}$ of $L \subseteq A^*$ is a union of classes of the partition $A^*/\equiv_L$. Indeed,

$$u^{-1}Lv^{-1} = \{\, w \in A^* \mid uwv \in L \,\} = \{\, w \in A^* \mid [u][w][v] \in F \,\}.$$

If we denote $m = [u]$ and $n = [v]$ then for the set $m^{-1}Fn^{-1} = \{\, s \in M \mid msn \in F \,\}$ we have

$$u^{-1}Lv^{-1} = \{\, w \in A^* \mid [u][w][v] \in F \,\} = \{\, w \in A^* \mid [w] \in m^{-1}Fn^{-1}\}. \qquad (*)$$

These basic observations lead us to an alternative description of the canonical biautomaton of the language $L$.

We denote $\mathcal{B}_L = (B_L, A, \cdot, \circ, i, T)$ where

- $B_L = \{m^{-1}Fn^{-1} \mid m, n \in M\}$;
- $m^{-1}Fn^{-1} \cdot a = [a]^{-1}(m^{-1}Fn^{-1}) = (m[a])^{-1}Fn^{-1}$;
- $m^{-1}Fn^{-1} \circ a = m^{-1}F([a]n)^{-1}$;
- $i = F = 1^{-1}F1^{-1}$;
- $T = \{m^{-1}Fn^{-1} \mid 1 \in m^{-1}Fn^{-1}\} = \{m^{-1}Fn^{-1} \mid mn \in F\}$.

**Proposition 3.9.** *Let $L$ be a regular language. Then $\mathcal{B}_L$ is a minimal biautomaton of the language $L$.*

*Proof.* It can be easily checked that $\mathcal{B}_L$ satisfies all conditions in the definition of a biautomaton. To prove the minimality it is enough to formalize the idea from the beginning of this subsection. Let $\zeta : C_L \to B_L$ be a mapping given by $\zeta(u^{-1}Lv^{-1}) = [u]^{-1}F[v]^{-1}$. This definition is correct because $[u]^{-1}F[v]^{-1} = \{[w] \mid w \in u^{-1}Lv^{-1}\}$ by $(*)$ and the value $[u]^{-1}F[v]^{-1}$ does not depend on the words $u$ and $v$ but only on the set $u^{-1}Lv^{-1}$. Now one can easily prove that $\zeta$ is an isomorphism from the canonical biautomaton $\mathcal{C}_L$ to the biautomaton $\mathcal{B}_L$. □

**Remark 3.10.** The previous construction can be also modified to an arbitrary surjective homomorphism $\xi : A^* \to N$, $u \mapsto [u]$ of monoids, and a subset $F \subseteq N$ recognizing the regular language $L = \{\, u \in A^* \mid [u] \in F \,\}$. We define the biautomaton $\mathcal{B}_\xi = (B_\xi, A, \cdot, \circ, i, T)$, where

- $B_\xi = N \times N$;
- for every $a \in A$ and $p, r \in N$; we set $(p, r) \cdot a = (p[a], r)$;
- similarly $(p, r) \circ a = (p, [a]r)$;
- $i = ([\lambda], [\lambda]) = (1, 1)$;
- $T = \{\, (p, r) \mid pr \in F \,\}$.

Now one can check that $\mathcal{B}_\xi$ is a biautomaton. Moreover, we see that the following five conditions are equivalent

$$u \in \mathscr{L}(\mathcal{B}_\xi); \quad ([\lambda], [\lambda]) \cdot u \in T; \quad ([u], [\lambda]) \in T; \quad [u] \in F; \quad u \in L.$$

Hence the constructed biautomaton $\mathcal{B}_\xi$ recognizes $L$.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

### 3.6. Biautomata and graphs

Let $G = (V, E)$ be an oriented graph, *i.e.* $V$ is a finite non-empty set of *vertices* and $E \subseteq V \times V$ is a set of *edges*. A sequence $(v_0, v_1, \ldots, v_n)$, $n \geq 2$, $v_n = v_0 \neq v_1$, of vertices, such that $(v_0, v_1), \ldots, (v_{n-1}, v_n) \in E$, is a *cycle* in $G$. The graph $V$ is *acyclic* if it contains no cycles. Note that "loops" are allowed in an acyclic graph.

On the set $V$ of vertices, we define the relation $\approx$ by $p \approx q$ if and only if there is a path from $p$ to $q$ and also a path from $q$ to $p$. Clearly, it is an equivalence relation on the set $V$ and the induced subgraphs on the classes of $V/\approx$ are called the *strongly connected components* (scc in short) of $G$.

We can also consider the *quotient* (oriented) graph $G/\approx$ having the scc's as vertices and there is an edge between different scc's $S$ and $S'$ if and only if there is an edge from $p \in S$ to $q \in S'$. Note that such a graph is always acyclic and contains no loops. For $v \in V$, we define the number $m(v)$ as the maximal length of a path in $G/\approx$ starting in the scc containing $v$.

For each biautomaton $\mathcal{B}$ we can consider its (oriented) graph

$$\mathsf{G}(\mathcal{B}) = (Q, \{(q, q \cdot a) \mid q \in Q,\ a \in A\} \cup \{(q, q \circ a) \mid q \in Q,\ a \in A\})$$

having *left* edges $(q, q \cdot a)$ and *right* edges $(q, q \circ a)$ (in case $q \cdot a = q \circ b$ the edge $(q, q \cdot a)$ is both left and right). Left (resp. right) corresponds here to reading from left (resp. right). A biautomaton $\mathcal{B}$ is *acyclic* if its graph $\mathsf{G}(\mathcal{B})$ is. Furthermore, for a biautomaton $\mathcal{B}$ with initial state $i$, we define the number $m(\mathcal{B})$ as $m(i)$ in the graph $\mathsf{G}(\mathcal{B})$.

## 4. Biautomata for piecewise testable languages

### 4.1. Proof of Theorem 1.1

A regular language $L$ over an alphabet $A$ is called *piecewise testable* if it is a Boolean combination of languages of the form $A^*a_1A^*a_2A^* \ldots A^*a_\ell A^*$, where $a_1, \ldots, a_\ell \in A$, $\ell \geq 0$. An effective characterization of piecewise testable languages was given by Simon [9, 10] who proved that a language $L$ is piecewise testable if and only if its syntactic monoid is $\mathcal{J}$-trivial. Here we give an alternative effective characterization of piecewise testable languages *via* biautomata.

For words $u, v \in A^*$, we write $u \lhd v$ if and only if $u$ is a subword of $v$, *i.e.* there are letters $a_1, \ldots, a_\ell \in A$ and words $v_0, v_1, \ldots, v_\ell \in A^*$ such that $u = a_1 \ldots a_\ell$ and $v = v_0a_1v_1 \ldots a_\ell v_\ell$. For $v \in A^*$, we denote $\mathsf{Sub}_k(v) = \{\, u \in A^+ \mid u \lhd v, |u| \leq k \,\}$. We define the equivalence relation $\sim_k$ on $A^*$ by the rule: $u \sim_k v$ if and only if $\mathsf{Sub}_k(u) = \mathsf{Sub}_k(v)$. Note that for $k = 1$, the set $\mathsf{Sub}_k(v)$ is equal to $\mathsf{c}(u)$. Further, for a given word $u \in A^*$, we denote by $L_u$ the language of all words which contain the word $u$ as a subword, *i.e.* $L_u = \{\, v \in A^* \mid u \lhd v \,\}$. If $u = a_1a_2 \ldots a_\ell$, where $a_1, a_2, \ldots, a_\ell \in A$, then we can write $L_u = A^*a_1A^*a_2A^* \ldots A^*a_\ell A^*$. An easy consequence of the definition of piecewise testable languages is the following lemma. The proof can be found *e.g.* in [4, 9]. In fact the proof is so easy that many

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

authors skip it and even in some papers the condition from the lemma is taken as a definition condition for piecewise testable languages.

**Lemma 4.1.** *A language $L$ is piecewise testable if and only if there exists an index $k$ such that $L$ is a union of classes in the partition $A^*/\sim_k$.*

Our goal is to prove Theorem 1.1, *i.e.* the characterization that the piecewise testable languages are exactly the languages with acyclic canonical biautomata.

**Example 4.2** (continuation of 3.7)**.** The biautomaton in Figure 1 is acyclic and therefore, by Theorem 1.1, the language $L$ is piecewise testable. In fact,

$$L = A^*cA^*aA^* \cap (A^*cA^*aA^*aA^*)^{\mathsf{c}} \cap (A^*cA^*cA^*aA^*)^{\mathsf{c}} \cap (A^*cA^*bA^*aA^*)^{\mathsf{c}}.$$

We need first some auxiliary statements.

**Lemma 4.3.** *Let $\mathcal{B}$ be an acyclic biautomaton and let $\sim$ be a congruence relation on $\mathcal{B}$. Then the quotient automaton $\mathcal{B}/\sim$ is acyclic.*

*Proof.* Let $([q_0]_\sim, [q_1]_\sim, \ldots, [q_n]_\sim)$ be a cycle in $\mathsf{G}(\mathcal{B}/\sim)$ with $[q_0]_\sim, \ldots, [q_{n-1}]_\sim$ pairwise different. We have $n \geq 2$, $q_n \sim q_0 \not\sim q_1$, and we can choose (possibly other representatives of the above classes of $\sim$) $q_1, \ldots, q_n$ and $a_1, \ldots, a_n \in A$ such that $q_0 *_1 a_1 = q_1$, $q_1 *_2 a_2 = q_2, \ldots, q_{n-1} *_n a_n = q_n$ where each $*_i$ is $\cdot$ or $\circ$. We can continue $q_n *_1 a_1 = q_{n+1}, \ldots, q_{2n-1} *_n a_n = q_{2n}$, $q_{2n} *_1 a_1 = q_{2n+1}, \ldots$. Notice that $q_i \sim q_j$ if and only if $i$ and $j$ are congruent modulo $n$, in particular $q_i \not\sim q_{i+1}$ and consequently $q_i \neq q_{i+1}$ for all $i \geq 0$. Since $\mathcal{B}$ is finite there are $i < j$ such that $q_i = q_j$.

Hence we have a cycle in $\mathsf{G}(\mathcal{B})$ starting at $q_i$. □

**Lemma 4.4.** *Let $L$ be a piecewise testable language over an alphabet $A$. Then the canonical biautomaton $\mathcal{C}_L = (C_L, A, \cdot, \circ, L, T)$ of $L$ is acyclic.*

*Proof.* Since every piecewise testable language over the alphabet $A$ is a Boolean combination of languages $L_u$, it is enough to prove the following:

**Claim 4.5.** $\mathcal{C}_{L_u}$ and $\mathcal{C}_{L_u^{\mathsf{c}}}$ are acyclic for every $u \in A^*$.

**Claim 4.6.** If $\mathcal{C}_K$ and $\mathcal{C}_L$ are acyclic then both $\mathcal{C}_{K\cap L}$ and $\mathcal{C}_{K\cup L}$ are also acyclic.

*Proof of Claim 4.5.* Indeed, for every $u \in A^*$, the canonical biautomaton $\mathcal{C}_{L_u} = (C_{L_u}, A, \cdot, \circ, L, T)$ of the language $L_u$ has states of the form $s^{-1}L_ut^{-1}$, where $s, t \in A^*$. Notice that for each word $w = b_1b_2 \ldots b_\ell$, where $b_1, b_2, \ldots, b_\ell \in A$, left derivatives of the language $L_w = A^*b_1A^*b_2A^* \ldots A^*b_\ell A^*$ are computed by the following rules: $b_1^{-1}L_w = L_{w'}$ where $w' = b_2 \ldots b_\ell$ and for $a \neq b_1$, $a \in A$, we have $a^{-1}L_w = L_w$. The same rules can be written for right derivatives and therefore each $s^{-1}L_ut^{-1}$ is of the form $L_v$, where $v \in A^*$ is a factor of $u$.

Let $v, w \in A^*$ and $a \in A$ be such that $L_w \neq L_v \in C_{L_u}$ and $L_w = L_v \cdot a$ or $L_w = L_v \circ a$. Then $|w| < |v|$ and we can deduce that the biautomaton $\mathcal{C}_{L_u}$ is acyclic. If we consider a language $L_u^{\mathsf{c}}$ instead of $L_u$ then the canonical biautomaton $\mathcal{C}_{L_u^{\mathsf{c}}}$ is acyclic because it is, in fact, the canonical biautomaton $\mathcal{C}_{L_u}$ where just terminal states are changed. □

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

*Proof of Claim 4.6.* Now if $K, L$ are languages such that $\mathcal{C}_K$ and $\mathcal{C}_L$ are acyclic then one can consider the direct product of biautomata $\mathcal{C}_K$ and $\mathcal{C}_L$ which is acyclic. In this structure we can choose, in the usual way, reachable states and also terminal states $T_{K\cap L}$ and $T_{K\cup L}$ respectively, namely $(p,q) \in T_{K\cap L}$ if and only if both $p$ and $q$ are terminal states in the biautomata $\mathcal{C}_K$ and $\mathcal{C}_L$ and $(p,q) \in T_{K\cup L}$ if and only if at least one of the states $p, q$ is terminal. In this way we obtain a certain acyclic biautomaton which recognizes the language $K \cap L$ (and $K \cup L$ respectively). To finish the proof, we can use Lemmas 3.8 and 4.3. □

Now we prove the difficult part of Theorem 1.1. The basic idea, namely reading one word from left and the other from right, is inspired by the first author's recent combinatorial proof [2] of Simon's result.

**Lemma 4.7.** *Let $L$ be a regular language such that the canonical biautomaton $\mathcal{C}_L$ of $L$ is acyclic. Then $L$ is a piecewise testable language.*

*Proof.* With respect to Lemma 4.1, we need to find an appropriate index $k$ such that $L$ is a union of some classes in the partition $A^*/\sim_k$. Such $k$ will be 2 times the size of the canonical biautomaton $\mathcal{C}_L = (C_L, A, \cdot, \circ, L, T)$ and the proof will be given by induction with respect to this $k$.

**Claim 4.8.** Let $\mathcal{B} = (B, A, \cdot, \circ, i, T)$ be an arbitrary acyclic biautomaton such that $|B| = \ell$. For every $u, v \in A^*$ such that $\mathsf{Sub}_{2\ell}(u) = \mathsf{Sub}_{2\ell}(v)$ and every $q \in B$, we have $q \cdot u \in T$ if and only if $q \cdot v \in T$.

*Proof.* For $\ell = 1$, the statement is trivial. Let $\ell > 1$ be arbitrary and assume that the statement holds for all smaller numbers. Let $q \in B$ be arbitrary and $u, v \in A^*$ be such that $\mathsf{Sub}_{2\ell}(u) = \mathsf{Sub}_{2\ell}(v)$. We will assume that $q \cdot u \in T$ and $q \cdot v \notin T$ and we show that this assumption leads to a contradiction. Recall that $q \cdot v \notin T$ is equivalent to $q \circ v \notin T$. In the state $q$ we read $u$ from left and $v$ from right and we are interested in the position in the words, where we leave the state $q$. First assume that $q \cdot u = q \in T$, *i.e.* we do not leave the state $q$. Then $\mathsf{Sub}_{2\ell}(u) = \mathsf{Sub}_{2\ell}(v)$ implies $\mathsf{c}(u) = \mathsf{c}(v)$ and we have $q \cdot v = q \in T$ – a contradiction. Thus $q \cdot u \neq q$ and in the same way we can show that $q \circ v \neq q$. Hence we really leave the state $q$ and there are $u', u'' \in A^*$, $a \in A$ such that $u = u'au''$, for every $c \in \mathsf{c}(u')$ we have $q \cdot c = q$, and $q \cdot a \neq q$. In particular $a \notin \mathsf{c}(u')$. Similarly, let $v', v'' \in A^*$, $b \in A$ be such that $v = v'bv''$, for every $c \in \mathsf{c}(v'')$ we have $q \circ c = q$, and $q \circ b \neq q$ (possibly $a = b$). Recall that we have $\mathsf{c}(u) = \mathsf{c}(v)$ and we can look for the first occurrence of $a$ in the word $v$ and the last occurrence of $b$ in the word $u$. We distinguish three cases depending on the relative positions of these occurrences of $a$ and $b$ in $u$. In general, note that for $x, y \in A$, $w \in A^*$, we have $xy \in \mathsf{Sub}_2(w)$ if and only if the first occurrence of $x$ in $w$ is before the last occurrence of $y$ in $w$. We will use this property together with $\mathsf{Sub}_2(u) = \mathsf{Sub}_2(v)$ which follows from the assumption $\mathsf{Sub}_{2\ell}(u) = \mathsf{Sub}_{2\ell}(v)$.

**Case I.** *The first occurrence of $a$ in $u$ is before the last occurrence of $b$ in $u$.* Since $\mathsf{Sub}_2(u) = \mathsf{Sub}_2(v)$ the same is true for $v$ and we can consider the

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

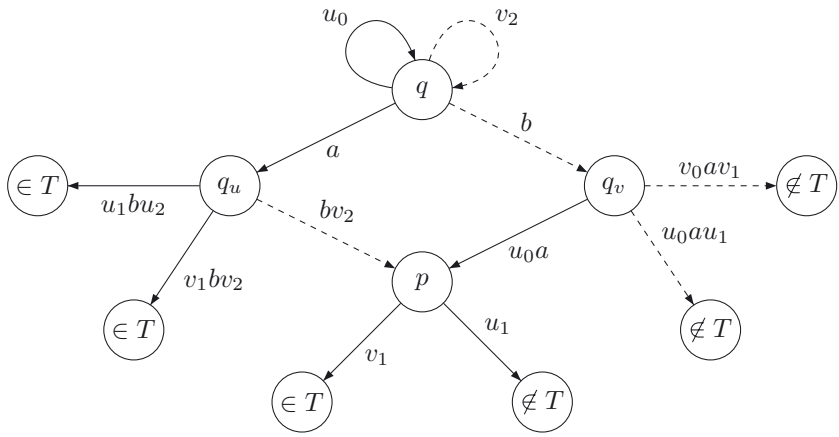



FIGURE 2. States in the proof of Case I from Lemma 4.7.

following decompositions of $u$ and $v$: $u = u_0au_1bu_2$, $v = v_0av_1bv_2$ where $u_0 = u', u_1, u_2, v_0, v_1, v_2 = v'' \in A^*$ are such that $a \notin \mathsf{c}(u_0)$, $a \notin \mathsf{c}(v_0)$, $b \notin \mathsf{c}(u_2)$, $b \notin \mathsf{c}(v_2)$. If we consider an arbitrary $w \in \mathsf{Sub}_{2\ell-1}(u_1bu_2)$, then $aw \in \mathsf{Sub}_{2\ell}(u) = \mathsf{Sub}_{2\ell}(v)$ from which $w \in \mathsf{Sub}_{2\ell-1}(v_1bv_2)$ follows. This means $\mathsf{Sub}_{2\ell-1}(u_1bu_2) \subseteq \mathsf{Sub}_{2\ell-1}(v_1bv_2)$ and the opposite inclusion can be proved in the same way. Thus we have $\mathsf{Sub}_{2\ell-1}(u_1bu_2) = \mathsf{Sub}_{2\ell-1}(v_1bv_2)$ and similarly $\mathsf{Sub}_{2\ell-1}(u_0au_1) = \mathsf{Sub}_{2\ell-1}(v_0av_1)$ and $\mathsf{Sub}_{2\ell-2}(u_1) = \mathsf{Sub}_{2\ell-2}(v_1)$.

The following part of the proof is illustrated in Figure 2.

Notice that the depicted states are not necessarily pairwise different. But the crucial property is that the state $q$ is different from all other ones.

We denote $q_u = q \cdot u_0a \neq q$ and we can consider the biautomaton consisting of all states reachable from $q_u$. This is an acyclic biautomaton with at most $\ell - 1$ states, because it is a subset of $B$ and it does not contain the state $q$. By induction assumption $q_u \cdot u_1bu_2 \in T$ if and only if $q_u \cdot v_1bv_2 \in T$. The first condition is satisfied because $q_u \cdot u_1bu_2 = q \cdot u_0au_1bu_2 = q \cdot u$. Hence $q_u \cdot v_1bv_2 = (q_u \cdot v_1) \cdot bv_2 \in T$ and also $(q_u \circ bv_2) \cdot v_1 = (q_u \cdot v_1) \circ bv_2 \in T$. We denote the state $q_u \circ bv_2$ as $p$.

Analogously, we denote $q_v = q \circ bv_2 = (q \circ v_2) \circ b \neq q$ and we consider the acyclic biautomaton consisting of all states reachable from $q_v$. We have $q_v \circ v_0av_1 = q \circ v \notin T$ hence $q_v \circ u_0au_1 \notin T$ follows from the induction assumption. Since we work with the biautomaton we deduce that $q_v \cdot u_0au_1 = (q_v \cdot u_0a) \cdot u_1 \notin T$. Now we can see that $q_v \cdot u_0a = (q \circ bv_2) \cdot u_0a = (q \cdot u_0a) \circ bv_2 = q_u \circ bv_2 = p$. We have observed $p \cdot v_1 \in T$ in the previous paragraph and $p \cdot u_1 \notin T$ here. It is clear that $p \neq q$ and we can consider the biautomaton consisting of all states reachable from $p$, which has at most $\ell - 1$ states. Since $\mathsf{Sub}_{2\ell-2}(u_1) = \mathsf{Sub}_{2\ell-2}(v_1)$ we see that $p \cdot v_1 \in T$ and $p \cdot u_1 \notin T$ cannot hold simultaneously. We obtain a contradiction.

**Case II.** *The first occurrence of $a$ in $u$ is also the last occurrence of $b$ in $u$.* In other words, $a = b$ and the first occurrence of $a$ is the unique occurrence of

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

this letter in $u$. Hence $a \in \mathsf{c}(u) = \mathsf{c}(v)$, $aa \notin \mathsf{Sub}_2(u) = \mathsf{Sub}_2(v)$ and $a$ has the unique occurrence in $v$ too. In the same manner as in Case I we can deduce that $\mathsf{Sub}_{2\ell-1}(u') = \mathsf{Sub}_{2\ell-1}(v')$ and $\mathsf{Sub}_{2\ell-1}(u'') = \mathsf{Sub}_{2\ell-1}(v'')$. In particular $\mathsf{c}(u') = \mathsf{c}(v')$ and $\mathsf{c}(u'') = \mathsf{c}(v'')$ which give $q \cdot v' = q$ and $q \circ u'' = q$. Now $q \cdot u = (q \cdot u'a) \cdot u'' \in T$ implies $(q \cdot u'a) \circ u'' \in T$, and thus $(q \cdot u'a) \circ u'' = (q \circ u'') \cdot u'a = q \cdot u'a = q \cdot a$. In the same way $q \circ v = (q \circ av'') \circ v' \notin T$ implies $(q \circ av'') \cdot v' \notin T$, and thus $(q \circ av'') \cdot v' = (q \cdot v') \circ av'' = q \circ av'' = (q \circ v'') \circ a = q \circ a$. We get $q \cdot a \in T$ and $q \circ a \notin T$ which is not possible in biautomata – a contradiction.

**Case III.** *The first occurrence of $a$ in $u$ is after the last occurrence of $b$ in $u$.* This means that $ab \notin \mathsf{Sub}_2(u) = \mathsf{Sub}_2(v)$ and the first occurrence of $a$ in $v$ is after the last occurrence of $b$ in $v$. We can consider the following decompositions of $u$ and $v$: $u = u_0bu_1au_2$, $v = v_0bv_1av_2$ where $u_0, u_1, u_2, v_0, v_1, v_2 \in A^*$ are such that $u_0bu_1 = u'$, $v_1av_2 = v''$. Again we can deduce that $\mathsf{Sub}_{2\ell-1}(u_0) = \mathsf{Sub}_{2\ell-1}(v_0)$ and $\mathsf{Sub}_{2\ell-1}(u_2) = \mathsf{Sub}_{2\ell-1}(v_2)$, in particular $\mathsf{c}(u_0) = \mathsf{c}(v_0)$ and $\mathsf{c}(u_2) = \mathsf{c}(v_2)$. Hence for every $c \in \mathsf{c}(u') = \mathsf{c}(u_0bu_1)$ we have $q \cdot c = q$, in particular $q \cdot b = q$ and $q \cdot c = q$ for every $c \in \mathsf{c}(u_0) = \mathsf{c}(v_0)$. Now we see that $q \circ v = q \circ v_0bv_1av_2 = (q \circ v_1av_2) \circ v_0b = q \circ v_0b \notin T$. Hence $q \cdot v_0b \notin T$, $q \cdot v_0b = q$ and we deduce $q \notin T$. On the other hand, for every $c \in \mathsf{c}(v'') = \mathsf{c}(v_1av_2)$ we have $q \circ c = q$, in particular $q \circ a = q$ and $q \circ c = q$ for every $c \in \mathsf{c}(v_2) = \mathsf{c}(u_2)$. Now $q \cdot u = (q \cdot u_0bu_1) \cdot au_2 = q \cdot au_2 \in T$. Hence $q \circ au_2 \in T$, $q \circ au_2 = q$. But this means $q \in T$ which contradicts the previous conclusion $q \notin T$.

We have proved the claim which completes the proof of the lemma. □

Now Theorem 1.1 is a consequence of Lemmas 4.4 and 4.7.

### 4.2. Simon's theorem as a consequence of Theorem 1.1

Recall Simon's famous result giving the effective characterization of piecewise testable languages.

**Result 1** (Simon [9,10])**.** Let $L$ be a language over a finite alphabet $A$. Then $L$ is piecewise testable if and only if the syntactic monoid $\mathsf{M}(L)$ is $\mathcal{J}$-trivial.

The statement consists of two implications where one of them is easy to prove. Namely, one can easily show that for each word $u \in A^*$ the syntactic monoid of the language $L_u$ is $\mathcal{J}$-trivial. Using standard algebraic considerations, this implies that every piecewise testable language has a $\mathcal{J}$-trivial syntactic monoid. Here we want to show the difficult implication in Simon's theorem as a consequence of Theorem 1.1.

**Lemma 4.9.** *Let $L$ be a language over a finite alphabet $A$ such that the syntactic monoid $M = \mathsf{M}(L)$ is $\mathcal{J}$-trivial. Then the canonical biautomaton $\mathcal{C}_L$ of $L$ is acyclic. Therefore $L$ is piecewise testable.*

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

*Proof.* Let $L$ be a regular language and $\eta$ its syntactic homomorphism onto $M$. We consider the biautomaton $\mathcal{B}_\eta$ from Remark 3.10. We claim that the biautomaton $\mathcal{B}_\eta$ is acyclic. Indeed, assume that $\mathsf{G}(\mathcal{B}_\eta)$ contains a cycle $(q_0, q_1, \dots, q_n)$, where $q_n = q_0 \neq q_1$, and letters $a_1, \dots, a_n \in A$ are such that for each $i = 1, \dots, n$ we have $q_{i-1} \cdot a_i = q_i$ or $q_{i-1} \circ a_i = q_i$. Assume additionally that $q_1 = q_0 \cdot a_1$. (The case $q_1 = q_0 \circ a_1$ can be treated dually). Then we have $q_0 = (p_0, r_0)$, where $p_0, r_0 \in M$ and $q_1 = (p_0[a_1], r_0)$. Thus $p_0[a_1] \neq p_0$. Now $q_n = q_0$ implies that there are $u, v \in A^*$ such that $(q_1 \cdot u) \circ v = q_0$. Hence $p_0[a_1][u] = p_0$. We have found two different elements $p_0[a_1]$ and $p_0$ which are $\mathcal{J}$-related. This is a contradiction to the assumption that $M$ is $\mathcal{J}$-trivial.

Finally, $\mathcal{C}_L$ can be obtained as a quotient biautomaton of the biautomaton $\mathcal{B}_\eta$ by Lemma 3.8. Hence $\mathcal{C}_L$ is acyclic by Lemma 4.3. □

## 5. Prefix-suffix testable languages, proof of Theorem 1.2

A language $L \subseteq A^*$ is *prefix-suffix testable* if it is a Boolean combination of the languages of the form $vA^*$, $A^*w$ where $v, w \in A^*$. Usually this terminology is used for languages of non-empty words (the so-called $+$-languages) – see Section 5.3 in [7] and references given there. We have chosen to work with $*$-languages to avoid several technical modifications in our theory of biautomata needed when considering $+$-languages. We present a full proof of the following proposition since we really need that result and it seems that the corresponding result in [7], namely Proposition 5.17, is not completely correct. For example, the prefix-suffix testable $+$-language $L = (aA^* \cap A^*b) \cup (bA^* \cap A^*a)$ over the alphabet $A = \{a, b\}$ cannot be written in the form mentioned in Proposition 5.17 of [7].

**Proposition 5.1.** *A language $L \subseteq A^*$ is prefix-suffix testable if and only if it is of the form*

$$v_1 A^* w_1 \cup \cdots \cup v_p A^* w_p \cup \{u_1, \dots, u_q\},$$

$$\text{where } p, q \geq 0,\ v_1, \dots v_p, w_1, \dots w_p, u_1, \dots, u_q, \in A^*.$$

*Proof.*

"⇒": Let $v \in A^+$. Then

$$(vA^*)^{\mathsf{c}} = v_1 A^* \cup \cdots \cup v_r A^* \cup \{u_1, \dots, u_s\}$$

where $v_1, \dots, v_r$ are all words of the length $|v|$ different from $v$ and $u_1, \dots, u_s$ are all words from $A^*$ which are shorter than $v$. Similarly, for $A^*w$, $w \in A^+$. Also $(A^*)^{\mathsf{c}} = \emptyset$ and so we do not need complements to express the considered languages.

Clearly, if $K = vA^* \cap wA^* \neq \emptyset$ then $v$ is a prefix of $w$ and $K = wA^*$ or $w$ is a prefix of $v$ and $K = vA^*$. Similarly for $A^*v \cap A^*w$. Further, for $v, w \in A^+$, we have

$$vA^* \cap A^*w = vA^*w \cup \{u_1, \dots, u_s\},$$

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

where $u_1, \ldots, u_s$ are all words of length less than $|v| + |w|$ with prefix $v$ and suffix $w$. Using both simple ideas from the above considerations, we get that

$$v_1 A^* w_1 \cap v_2 A^* w_2 = v' A^* w' \cup \{u'_1, \ldots, u'_t\},$$

for appropriate $v', w', u'_1, \ldots, u'_t$ or it is the empty set.

"$\Leftarrow$": Clearly, for $u \in A^*$, we can write $\{u\} = uA^* \cap (ua_1A^*)^{\mathsf{c}} \cap \cdots \cap (ua_nA^*)^{\mathsf{c}}$ where $A = \{a_1, \ldots, a_n\}$.

Finally, for $v, w \in A^*$, the equality $vA^*w = (vA^* \cap A^*w) \setminus \{u_1, \ldots, u_s\}$ holds, where $u_1, \ldots, u_s$ are again all words of length less than $|v| + |w|$ with prefix $v$ and suffix $w$. □

Note that the expression from the last proposition is not uniquely determined.

The following result ensures that all states of the canonical biautomaton of a prefix-suffix testable language are also prefix-suffix testable languages.

**Lemma 5.2.** *The class of all prefix-suffix testable languages is closed under taking derivatives.*

*Proof.* Let $u, v \in A^*$ and consider $L = a^{-1}uA^*v$. If $u = a \cdot u'$ then $L = u'A^*v$. If $u$ is not the empty word $\lambda$ and the first letter in $u$ is different from $a$, then $L = \emptyset$. Furthermore, if $u = \lambda$ and $v = av'$ then $L = A^*v \cup \{v'\}$, and if $u = \lambda$ and $v$ is not of the form $av'$ then $L = A^*v$. We can establish similar equalities for right derivatives. By Proposition 5.1 we get the statement. □

The following result is one of the implications in Theorem 1.2.

**Lemma 5.3.** *Let $L \subseteq A^*$ be a prefix-suffix testable language. Then the canonical biautomaton $\mathcal{C}_L$ of $L$ satisfies the condition* (†).

*Proof.* We will work with expressions which are formal finite sums of finite words and terms of the form $vA^*w$ where $v, w$ are words from $A^*$. For such an expression

$$E = v_1 A^* w_1 + \cdots + v_p A^* w_p + u_1 + \cdots + u_q$$

where $p, q \geq 0$, $v_1, \ldots v_p, w_1, \ldots w_p, u_1, \ldots, u_q \in A^*$, we denote the corresponding prefix-suffix testable language

$$v_1 A^* w_1 \cup \cdots \cup v_p A^* w_p \cup \{u_1, \ldots, u_q\}$$

by $\mathcal{L}(E)$. For the expression $E$ we define $\ell(E)$ as the maximum of the lengths of the words $v_1, \ldots, v_p$. Similarly we define $r(E)$ of the expression $E$ as the maximum of the lengths of the words $w_1, \ldots, w_p$. Finally, we put $s(E) = \ell(E) + r(E)$.

Now, for a letter $a \in A$, we define an operation $a^{-1}$ on the set of all considered expressions given by the rules:

- for expressions $E_1, E_2$ we put $a^{-1}(E_1 + E_2) = a^{-1}E_1 + a^{-1}E_2$;
- for $u = au' \in A^*$ we put $a^{-1}u = u'$;

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

- for $u \in A^*$ which is not of the form $au'$, we put $a^{-1}u = \emptyset$;
- for $w \in A^*$ we put $a^{-1}(A^*w) = A^*w + a^{-1}w$;
- for $v, w \in A^*$, $v \neq \lambda$ we put $a^{-1}(vA^*w) = (a^{-1}v)A^*w$.

Now by the equalities in the proof of Lemma 5.2, for every expression $E$, we have $\mathcal{L}(a^{-1}E) = a^{-1}\mathcal{L}(E)$. Moreover, $\ell(a^{-1}E) \leq \ell(E)$ and the equality holds here if and only if $\ell(E) = 0$. On the other side, $r(a^{-1}E) = r(E)$. We can also define $u^{-1}E$ inductively with respect to the length of a word $u \in A^*$ and observe that $\mathcal{L}(u^{-1}E) = u^{-1}\mathcal{L}(E)$, $r(u^{-1}E) = r(E)$, $\ell(u^{-1}E) \leq \ell(E)$ and $\ell(u^{-1}E) = \ell(E)$ if and only if $\ell(E) = 0$. We can define $Ea^{-1}$ and $Eu^{-1}$ in the same way and we can observe similar properties.

After the preliminary considerations above, now let $L$ be a prefix-suffix testable language. By the definition of the canonical biautomaton $\mathcal{C}_L$ and since prefix-suffix testable languages are closed under derivatives, every state $q$ in $\mathcal{C}_L$ is in fact a prefix-suffix testable language $K$. Assume that there are words $u, v \in A^+$ such that $q \cdot u = q \circ v = q$, *i.e.* $u^{-1}K = Kv^{-1} = K$. Among all expressions $E$ with the property $\mathcal{L}(E) = K$ we choose some with minimal $s(E)$. We claim that $s(E) = 0$. Indeed, if $\ell(E) \neq 0$ then $\ell(u^{-1}E) < \ell(E)$ which is in contradiction with the equality $u^{-1}K = K$. Thus $\ell(E) = 0$ and also $r(E) = 0$ by the dual argument. Hence $E$ is a finite sum of words or $E$ is an expression of the form $A^* + U$, where $U$ is a finite sum of words. In the first case, it is not possible to have $u^{-1}K = K$ because these two finite languages have different lengths of the longest words. In the second case, we have $K = \mathcal{L}(A^* + U) = A^*$ from which $K \cdot a = a^{-1}K = a^{-1}A^* = A^* = K$ and similarly $K \circ a = Ka^{-1} = K$ follows. This means that $q = K$ is an absorbing state in the biautomaton $\mathcal{C}_L$. □

For considered languages, the strongly connected components of the corresponding graphs $\mathsf{G}(\mathcal{C}_L)$ are of very special forms as stated below.

**Lemma 5.4.** *Let $L \subseteq A^*$ be a language and let $\mathcal{C}_L$ be its canonical biautomaton satisfying the condition (†). Then each strongly connected component $S$ of $\mathsf{G}(\mathcal{C}_L)$ is of one of the following types:*

1. *a single vertex (no loops are allowed);*
2. *there are edges in $S$ and all of them are left;*
3. *there are edges in $S$ and all of them are right;*
4. *$S$ consists of a single absorbing state.*

*In particular, for each $q \in S$ and $a \in A$, the edge $(q, q \cdot a)$ runs outside of $S$ in cases* (1) *and* (3)*, and the edge $(q, q \circ a)$ runs outside of $S$ in cases* (1) *and* (2)*.*

*Proof.* First, we will prove the following consequence of (†):

$$\forall\, q \in C_L,\ u, v \in A^+,\ \text{the fact } (q \cdot u) \circ v = q \text{ implies that } q \text{ is absorbing.} \qquad (\dagger')$$

For this purpose choose $q \in C_L$ and $u, v \in A^+$ such that $(q \cdot u) \circ v = q$. Let $q_i = q \cdot u^i$ for each $i \geq 0$. For $i \geq 1$, we have

$$q_i \circ v = (q \cdot u^i) \circ v = ((q \cdot u) \circ v) \cdot u^{i-1} = q \cdot u^{i-1} = q_{i-1}.$$

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

Since $C_L$ is finite there exist $i < j$ such that $q_i = q_j$. Then we have $q_i \cdot u^{j-i} = q_j = q_i$ and $q_i \circ v^{j-i} = q_j \circ v^{j-i} = q_i$. By (†), the state $q_i$ is absorbing and since $q_i = q_i \circ v^i = q_0 = q$ the state $q$ is absorbing, which proves (†′).

Now having a scc $S$ with at least 2 states and with both left and right edges we can form a cycle $(p_0, \ldots, p_n)$ with both kinds of edges. We can transform this cycle repeatedly in such a way that each part $(p_i, p_{i+1}, p_{i+2})$, for $i \in \{0, \ldots, n-2\}$, such that $p_i \circ b = p_{i+1}$, $p_{i+1} \cdot c = p_{i+2}$, $b, c \in A$, is replaced by $(p_i, p_i \cdot c, p_{i+2})$ – here we use the condition from the definition of the biautomaton $(p \circ b) \cdot c = (p \cdot c) \circ b$. This procedure constructs the words $u, v \in A^+$ such that $(p_0 \cdot u) \circ v = p_0$. By (†′), $p_0$ is an absorbing state – a contradiction.

Finally a scc with a single vertex and both kinds of edges is absorbing by (†). □

Recall the definition of the invariant $m$ from Section 3.6.

**Lemma 5.5.** *Let $L \subseteq A^*$ be a language and let $\mathcal{C}_L$ be its canonical biautomaton satisfying the condition* (†). *Then, for each $u \in L$, $|u| \geq m(\mathcal{C}_L)$ there are $v, w \in A^*$ such that $u \in vA^*w \subseteq L$ and $|v| + |w| \leq m(\mathcal{C}_L)$.*

*Proof.* Notice that each state of $\mathcal{C}_L$ is of the form $K = v^{-1}Lw^{-1}$, $v, w \in A^*$, and $\mathcal{C}_K$ is a substructure of $\mathcal{C}_L$ and it satisfies the condition (†) again.

The statement of our lemma can be obtained from the following claim putting $q = i$.

**Claim 5.6.** For each $q \in C_L$ and $u = a_1 \ldots a_\ell$, $a_1, \ldots, a_\ell \in A$, $\ell \geq m(q) = k$ such that $q \cdot u \in T$ there exists $v, w \in A^*$ such that $u \in vA^*w \subseteq \mathscr{L}(\mathcal{C}_L, q)$ and $|v| + |w| \leq k$.

*Proof.* We will use induction, with respect to $k$, and Lemma 5.4 which describes the possible forms of scc's of $\mathsf{G}(\mathcal{C}_L)$.

If $q$ is in a scc of type (4), then $q$ is absorbing and we have $u \in A^* = \mathscr{L}(\mathcal{C}_L, q)$. In particular, we can use this observation for $k = 0$, which starts our induction.

Induction step: Let $k \geq 1$ and suppose the validity of our assertion for all states $p$ such that $m(p) < k$. Let $u = a_1 \ldots a_\ell$, where $a_1, \ldots, a_\ell \in A$, $\ell \geq k$.

If $q$ is in a scc of type (1) or (3) then $q \cdot a_1$ is not in the same scc as $q$ by Lemma 5.4 and $(q \cdot a_1) \cdot a_2 \ldots a_\ell \in T$. Thus $m(q \cdot a_1) < k$ and by the induction assumptions there exist $v, w \in A^*$ with $|v| + |w| < k$ such that $a_2 \ldots a_\ell \in vA^*w \subseteq \mathscr{L}(\mathcal{C}_L, q \cdot a_1)$. Then $u \in a_1 vA^*w \subseteq \mathscr{L}(\mathcal{C}_L, q)$.

Similarly, if $q$ is in a scc of type (2) then $q \circ a_\ell$ is not in the same scc as $q$ by Lemma 5.4 and $(q \circ a_\ell) \cdot a_1 \ldots a_{\ell-1} \in T$. Thus $m(q \circ a_\ell) < k$ and by the induction assumptions there exist $v, w \in A^*$ with $|v| + |w| < k$ such that $a_1 \ldots a_{\ell-1} \in vA^*w \subseteq \mathscr{L}(\mathcal{C}_L, q \circ a_\ell)$. Hence, $u \in vA^*wa_\ell \subseteq \mathscr{L}(\mathcal{C}_L, q)$. □

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press

*Proof of Theorem 1.2.* "⇐": Let $U$ be the set of all words from $L$ of length less than $m(\mathcal{C}_L)$. The language $L$ can be written in the form of Proposition 5.1 where $\{u_1, \ldots, u_q\} = U$ and the terms $vA^*w$ are those from Lemma 5.5. Since there are only finitely many possible pairs $(v, w)$ such that $|v| + |w| \leq m(\mathcal{C}_L)$, the statement follows.

"⇒": This implication is the content of Lemma 5.3. □

## 6. Conclusions

We have introduced the notion of biautomata; in particular we attach to each regular language $L$ its canonical biautomaton. Using graph properties of such structures we can decide a membership of $L$ in certain prominent classes of languages, namely piecewise testable languages and prefix-suffix testable ones. A future research could concentrate to get similar characterizations for further significant classes of regular languages. Other possibility is to consider "varieties" of biautomata and relate them to classes of languages *via* an Eilenberg-type theorem. Moreover, it could be interesting to consider a non-deterministic version of biautomata.

*Acknowledgements.* The authors would like to express their gratitude to Michal Kunc who actively participated in several discussions on the subject.

## References

[1] J. Brzozowski, Derivatives of regular expressions. *J. ACM* **11** (1964) 481–494.
[2] O. Klíma, Piecewise testable languages *via* combinatorics on words. *Disc. Math.* **311** (2011) 2124–2127.
[3] O. Klíma and L. Polák, On varieties of meet automata. *Theoret. Comput. Sci.* **407** (2008) 278–289.
[4] O. Klíma and L. Polák, Hierarchies of piecewise testable languages. *Int. J. Found. Comput. Sci.* **21** (2010) 517–533.
[5] S. Lombardy and J. Sakarovich, The universal automaton, in *Logic and Automata: History and Perspectives*, edited by J. Flum, E. Grödel and T. Wilke. Amsterdam University Press (2007) 457–504.
[6] J.-E. Pin, *Varieties of Formal Languages*. North Oxford, London and Plenum, New York (1986).
[7] J.-E. Pin, Syntactic semigroups, in *Handbook of Formal Languages*, Chap. 10, edited by G. Rozenberg and A. Salomaa. Springer (1997).
[8] L. Polák, Syntactic semiring and universal automata, in *Proc. of DLT 2003. Lect. Notes Comput. Sci.* **2710** (2003) 411–422.
[9] I. Simon, *Hierarchies of events of dot-depth one*. Ph.D. thesis. University of Waterloo (1972).
[10] I. Simon, Piecewise testable events, in *Proc. of ICALP 1975. Lect. Notes Comput. Sci.* **33** (1975) 214–222.

Communicated by M. Holzer.
Received November 11, 2011. Accepted May 3, 2012.

https://doi.org/10.1051/ita/2012014 Published online by Cambridge University Press