

Boolean-type retractable state-finite automata without outputs

By MÁRK FÜZESDI (Budapest)

Abstract. An automaton \mathbf{A} is called a retractable automaton if, for every subautomaton \mathbf{B} of \mathbf{A} , there is at least one homomorphism of \mathbf{A} onto \mathbf{B} which leaves the elements of B fixed (such homomorphism is called a retract homomorphism of \mathbf{A} onto \mathbf{B}). We say that a retractable automaton $\mathbf{A}=(A,X,\delta)$ is Boolean-type if there exists a family $\{\lambda_B \mid \mathbf{B}$ is a subautomaton of $\mathbf{A}\}$ of retract homomorphisms λ_B of \mathbf{A} such that, for arbitrary subautomata \mathbf{B}_1 and \mathbf{B}_2 of \mathbf{A} , the condition $B_1 \subseteq B_2$ implies $\text{Ker } \lambda_{B_2} \subseteq \text{Ker } \lambda_{B_1}$. In this paper, we describe the Boolean-type retractable state-finite automata without outputs.

1. Introduction and motivation

Let $\mathbf{A} = (A, X, \delta)$ be an automaton without outputs. A subautomaton \mathbf{B} of \mathbf{A} is called a retract subautomaton if there is a homomorphism of \mathbf{A} onto \mathbf{B} which leaves the elements of B fixed. A homomorphism with this property is called a retract homomorphism of \mathbf{A} onto \mathbf{B} .

In [5], A. NAGY introduced the notion of the retractable automaton. An automaton \mathbf{A} (without outputs) is called a retractable automaton if every subautomaton of \mathbf{A} is a retract subautomaton. In [5, Theorem 3], he proved that if the lattice $\mathcal{L}(\mathbf{A})$ of all congruences of an automaton \mathbf{A} is complemented, then \mathbf{A} is a retractable automaton. He also defined the notion of the Boolean-type retractable automaton. We say that a retractable automaton $\mathbf{A}=(A,X,\delta)$ is Boolean-type if there exists a family $\{\lambda_B \mid \mathbf{B}$ is a subautomaton of $\mathbf{A}\}$ of retract homomorphisms λ_B of \mathbf{A} such that, for arbitrary subautomata \mathbf{B}_1 and \mathbf{B}_2 of \mathbf{A} , the condition $B_1 \subseteq B_2$ implies $\text{Ker } \lambda_{B_2} \subseteq \text{Ker } \lambda_{B_1}$. In [5, Theorem 5], he proved that if the

Mathematics Subject Classification: 68Q70.

Key words and phrases: automaton, congruences of automaton.

lattice $\mathcal{L}(\mathbf{A})$ of all congruences of an automaton \mathbf{A} is a Boolean algebra, then \mathbf{A} is a Boolean-type retractable automaton.

In [5], A. NAGY investigated the not necessarily state-finite Boolean-type retractable automata containing traps (a state c is called a trap of an automaton $\mathbf{A}=(A,X,\delta)$ if $\delta(c,x) = c$ for every $x \in X$). He proved that every Boolean-type retractable automaton containing traps has a homomorphic image which is a Boolean-type retractable automaton containing exactly one trap. Moreover, he gave a complete description of Boolean-type retractable automata containing exactly one trap.

In [2], the authors defined the notion of the strongly retract extension of automata. They proved that every state-finite Boolean-type retractable automaton without outputs is a direct sum of Boolean-type retractable automata whose principal factors form a tree. Moreover, a state-finite automaton \mathbf{A} is a Boolean-type retractable automaton, whose principal factors form a tree if and only if it is a strongly retract extension of a strongly connected subautomaton of \mathbf{A} by a Boolean-type retractable automaton containing exactly one trap (which is described in [5]).

In [5] and [2], some theorems give only necessary conditions for special retractable or Boolean-type retractable state-finite automata without outputs. Paper [6] is the first to provide a complete description of state-finite retractable automata without outputs. Using the results of [6], we give a complete description of Boolean-type retractable state-finite automata without outputs.

2. Basic notations

By an automaton without outputs we mean a system (A, X, δ) where A and X are non-empty sets, and δ maps from the Cartesian product $A \times X$ to A . We will refer to A , X and δ as the state set, the input set and the transition function of \mathbf{A} , respectively. An automaton \mathbf{A} is said to be state-finite, if the set A is finite. In this paper by an automaton we always mean a state-finite automaton without outputs. We note that if the state set is finite, then there are only finite-many transformations of the state set generated by the elements of the input set. An obvious consequence of this fact is that we get the same results of this paper considering finite automata (without outputs) instead of state-finite automata (without outputs). We will follow the definitions and notations of [6].

An automaton $\mathbf{B}=(B,X,\delta_B)$ is called a subautomaton of an automaton $\mathbf{A}=(A,X,\delta)$ if B is a subset of A , and δ_B is the restriction of δ to $B \times X$.

A subautomaton \mathbf{B} of an automaton \mathbf{A} contained by every subautomaton of \mathbf{A} is called the kernel of \mathbf{A} .

By a homomorphism of an automaton (A, X, δ) into an automaton (B, X, γ) we mean a map ϕ of the set A into the set B such that $\phi(\delta(a, x)) = \gamma(\phi(a), x)$, for all $a \in A$ and $x \in X$.

A congruence of an automaton (A, X, δ) is an equivalence α of the set A such that, for every states $a, b \in A$ and every input sign $x \in X$, the assumption $(a, b) \in \alpha$ implies $(\delta(a, x), \delta(b, x)) \in \alpha$. A congruence class α containing $a \in A$ will be denoted by $[a]_\alpha$. The kernel of a homomorphism $\phi : (A, X, \delta) \mapsto (B, X, \gamma)$, which is denoted by $\text{Ker } \phi$, is defined as the following relation on A : $\text{Ker } \phi := \{(a, b) \in A \times A : \phi(a) = \phi(b)\}$. It is clear that $\text{Ker } \phi$ is a congruence on A .

We will denote the lattice of all congruences of an automaton \mathbf{A} by $\mathcal{L}(A)$. For every $\alpha, \beta \in \mathcal{L}(A)$, let $\alpha \wedge \beta := \alpha \cap \beta$ and $\alpha \vee \beta = (\alpha \cup \beta)^T$, where

$$(\alpha \cup \beta)^T = (\alpha \cup \beta) \cup ((\alpha \cup \beta) \circ (\alpha \cup \beta)) \cup \dots$$

is the transitive closure of $\alpha \cup \beta$ (here \circ denotes the usual operation on the semigroup of all binary relations on A , see [3]).

Let $\mathbf{B}=(B,X,\delta_B)$ be a subautomaton of an automaton $\mathbf{A}=(A, X, \delta)$. The relation

$$\varrho_B = \{(b_1, b_2) \in A \times A : b_1 = b_2 \text{ or } b_1, b_2 \in B\}$$

is a congruence on \mathbf{A} . This congruence is called the Rees congruence on \mathbf{A} defined by \mathbf{B} . The ϱ_B -classes of A are B itself and every one-element set $\{a\}$ with $a \in A \setminus B$.

3. Retractable automata

Definition 1. A subautomaton \mathbf{B} of an automaton $\mathbf{A}=(A, X, \delta)$ is called a retract subautomaton if there exists a homomorphism λ_B of \mathbf{A} onto \mathbf{B} which leaves the elements of B fixed. An automaton is said to be retractable if its every subautomaton is retract ([5]).

Theorem 1. A Rees-congruence ϱ_B defined by a subautomaton $\mathbf{B}=(B,X,\delta_B)$ of an automaton $\mathbf{A}=(A, X, \delta)$ has a complement in the lattice $(\mathcal{L}(A), \vee, \wedge)$ if and only if \mathbf{B} is a retract subautomaton.

PROOF. Let $\mathbf{A}=(A, X, \delta)$ be an automaton. Assume that \mathbf{B} is a subautomaton of \mathbf{A} such that the Rees congruence ϱ_B has a complement in $\mathcal{L}(A)$. By the proof of [5, Theorem 3], \mathbf{B} is a retract subautomaton of \mathbf{A} .

Conversely, assume that \mathbf{B} is a retract subautomaton of \mathbf{A} . We will show that the kernel $\text{Ker } \lambda_B$ of a retract homomorphism λ_B of \mathbf{A} onto \mathbf{B} is the complement of the Rees congruence ϱ_B on \mathbf{A} defined by \mathbf{B} . We show this by proving that, for every states $a \neq b$ of \mathbf{A} , we have $(a, b) \notin \eta_B \wedge \varrho_B$ and $(a, b) \in \eta_B \vee \varrho_B$. Let a, b be arbitrary elements in A with the condition $a \neq b$.

Case $(a, b \in B)$. Then $(a, b) \notin \eta_B \Rightarrow (a, b) \notin \eta_B \cap \varrho_B = \eta_B \wedge \varrho_B$. Furthermore, $a\varrho_B b \Rightarrow (a, b) \in \varrho_B \cup \eta_B \subseteq \varrho_B \vee \eta_B$.

Case $(a \in A \setminus B, b \in B)$. In this case $(a, b) \notin \varrho_B$, and so $(a, b) \notin \eta_B \cap \varrho_B = \eta_B \wedge \varrho_B$.

- If $\lambda_B(a) = \lambda_B(b)$, that is, $(a, b) \in \eta_B$, then $(a, b) \in \eta_B \cup \varrho_B \subseteq \eta_B \vee \varrho_B$.
- If $\lambda_B(a) \neq \lambda_B(b)$, then there is a $c \in B$ such that $\lambda_B(a) = \lambda_B(c)$, that is, $(a, c) \in \eta_B \subseteq \varrho_B \cup \eta_B$. As $(c, b) \in \varrho_B \subseteq \varrho_B \cup \eta_B$, we have $(a, b) \in (\varrho_B \cup \eta_B) \circ (\varrho_B \cup \eta_B) \subseteq \varrho_B \vee \eta_B$.

Case $(a \in B, b \in A \setminus B)$. This case is similar to the previous case.

Case $(a, b \in A \setminus B)$. In this case $(a, b) \notin \varrho_B$, and so $(a, b) \notin \varrho_B \cap \eta_B = \varrho_B \wedge \eta_B$. Since λ_B maps A onto B and leaves the elements of B fixed, there are elements $c, d \in B$ such that $\lambda_B(a) = c = \lambda_B(c)$ and $\lambda_B(b) = d = \lambda_B(d)$. Then $(a, c) \in \eta_B \subseteq \varrho_B \cup \eta_B$, $(c, d) \in \varrho_B \subseteq \varrho_B \cup \eta_B$, $(d, b) \in \eta_B \subseteq \varrho_B \cup \eta_B$, and so $(a, b) \in (\varrho_B \cup \eta_B) \circ (\varrho_B \cup \eta_B) \circ (\varrho_B \cup \eta_B) \subseteq \varrho_B \vee \eta_B$. \square

4. Boolean-type retractable automata

Definition 2. We say that a retractable automaton $\mathbf{A} = (A, X, \delta)$ is Boolean-type if there exists a family $\{\lambda_B \mid \mathbf{B} \text{ is a subautomaton of } \mathbf{A}\}$ of retract homomorphism λ_B of \mathbf{A} such that, for arbitrary subautomata \mathbf{B}_1 and \mathbf{B}_2 of \mathbf{A} , the condition $B_1 \subseteq B_2$ implies $\text{Ker } \lambda_{B_2} \subseteq \text{Ker } \lambda_{B_1}$.

In the next, if we suppose that \mathbf{A} is a Boolean-type retractable automaton and \mathbf{C} is a subautomaton of \mathbf{A} , then λ_C will denote the retract homomorphism of \mathbf{A} onto \mathbf{C} belonging to a fix family $\{\lambda_B \mid \mathbf{B} \text{ is a subautomaton of } \mathbf{A}\}$ of retract homomorphisms λ_B of \mathbf{A} satisfying the conditions of Definition 2.

In this section, we shall discuss Boolean-type retractable state-finite automata without outputs. We describe these automata using the concepts and constructions of [6].

Definition 3. We say that an automaton $\mathbf{A} = (A, X, \delta)$ is a direct sum of subautomata $\mathbf{A}_i = (A_i, X, \delta_i)$ ($i \in I$) if $A_i \cap A_j = \emptyset$ for every $i, j \in I$ with $i \neq j$, and, moreover, $A = \bigcup_{i \in I} A_i$.

Theorem 2 ([6]). *For a state-finite automaton $\mathbf{A} = (A, X, \delta)$, the following statements are equivalent:*

- (i) \mathbf{A} is retractable.
- (ii) \mathbf{A} is the direct sum of finitely many state-finite retractable automata, which contain kernels isomorphic to each other.

The next lemma will be used in the proof of Theorem 3 several times.

Lemma 1. *If $\mathbf{D} \subseteq \mathbf{B}$ are subautomata of a Boolean-type retractable automaton \mathbf{A} such that $\lambda_B(a) \in D$ for some $a \in A$, then $\lambda_B(a) = \lambda_D(a)$.*

PROOF. Let $c = \lambda_B(a)$. As $c \in D \subseteq B$, we have $\lambda_B(c) = c$. Thus a and c are in the same $\text{Ker } \lambda_B$ -class of A . As every $\text{Ker } \lambda_B$ -class is in a $\text{Ker } \lambda_D$ -class, we have that a and c are in the same $\text{Ker } \lambda_D$ -class, and so $\lambda_D(a) = \lambda_D(c)$. As $c \in D$, we have $\lambda_D(c) = c$, and so $\lambda_D(a) = \lambda_D(c) = c = \lambda_B(a)$. \square

Theorem 3. *For a state-finite automaton $\mathbf{A} = (A, X, \delta)$, the following statements are equivalent:*

- (i) \mathbf{A} is a Boolean-type retractable automaton.
- (ii) \mathbf{A} is the direct sum of finitely many state-finite Boolean-type retractable automata containing kernels isomorphic to each other.

PROOF. (i) \mapsto (ii): Let \mathbf{A} be a Boolean-type retractable state-finite automaton. By Theorem 2, \mathbf{A} is a direct sum of finitely many state-finite retractable automata \mathbf{A}_i ($i \in I$) containing kernels isomorphic to each other. We show that \mathbf{A}_i is a Boolean-type retractable automaton for every $i \in I$. If \mathbf{C} is a subautomaton of \mathbf{A}_i ($i \in I$), then \mathbf{C} is a subautomaton of \mathbf{A} . Let $\lambda_{C,i}$ be the restriction of the retract homomorphism λ_C to A_i , where λ_C denotes the retract homomorphism of \mathbf{A} onto \mathbf{C} , belonging to a fixed family $\{\lambda_B \mid \mathbf{B} \text{ is a subautomaton of } \mathbf{A}\}$ of retract homomorphisms of \mathbf{A} satisfying conditions of Definition 2. It is clear that $\{\lambda_{C,i} \mid \mathbf{C} \text{ is a subautomaton of } \mathbf{A}_i\}$ is a family of retract homomorphisms of \mathbf{A}_i which satisfies conditions of Definition 2. Thus, \mathbf{A}_i is a Boolean-type retractable automaton for every $i \in I$.

(ii) \mapsto (i): Assume that the automaton \mathbf{A} is a direct sum of Boolean-type retractable automata \mathbf{A}_i ($i \in I = \{1, 2, \dots, n\}$) whose kernels \mathbf{T}_i are isomorphic to each other. Let $(\cdot)\varphi_{i,i}$ denote the identical mapping of T_i ($i = 1, \dots, n$). For

arbitrary $i = 1, \dots, n-1$, let $(\cdot)\varphi_{i,i+1}$ denote the corresponding isomorphism of T_i onto T_{i+1} . For arbitrary $i, j \in I$ with $i < j$, let $(\cdot)\Phi_{i,j} = \varphi_{i,i+1} \circ \dots \circ \varphi_{j-1,j}$. For arbitrary $i, j \in I$ with $i > j$, let $(\cdot)\Phi_{i,j} = \Phi_{j,i}^{-1} = \varphi_{i-1,i}^{-1} \circ \dots \circ \varphi_{j,j+1}^{-1}$. It is clear that $\Phi_{i,j}$ is an isomorphism of \mathbf{T}_i onto \mathbf{T}_j for every $i, j \in I$. Moreover, for every $i, j, k \in I$, $\Phi_{i,j} \circ \Phi_{j,k} = \Phi_{i,k}$.

Let \mathbf{B} be a subautomaton of \mathbf{A} . Let \mathcal{B} denote the set of all indexes i from $1, 2, \dots, n$ which satisfy $B_i = B \cap A_i \neq \emptyset$. If $i \in \mathcal{B}$, then $T_i \subseteq B_i$. Let $i_B = \min \mathcal{B}$.

We give a retract homomorphism Λ_B of \mathbf{A} onto \mathbf{B} . If $i \in \mathcal{B}$, then let $\Lambda_B(a) = \lambda_{B_i}(a)$ for every $a \in A_i$. If $i \in I \setminus \mathcal{B}$ (that is, $B_i = \emptyset$), then, for every $a \in A_i$, let $\Lambda_B(a) = (\lambda_{T_i}(a))\Phi_{i,i_B}$. It is a matter of checking to see that Λ_B is a retract homomorphism of \mathbf{A} onto \mathbf{B} .

We show that the set $\{\Lambda_B \mid \mathbf{B} \text{ is a subautomaton of } \mathbf{A}\}$ satisfies the conditions of Definition 2. Let $\mathbf{D} \subseteq \mathbf{B}$ be arbitrary subautomata of \mathbf{A} . We note that $\mathcal{D} \subseteq \mathcal{B}$ and $i_B \leq i_D$. Assume $(a, b) \in \text{Ker } \Lambda_B$ for some $a, b \in A$ (with $a \in A_i, b \in A_j$). Then

$$\Lambda_B(a) = \Lambda_B(b).$$

Case 1. $i \in \mathcal{D}$. In this case $i_D \leq i$. We have two subcases. If $j \in \mathcal{B}$, then

$$\lambda_{B_i}(a) = \Lambda_B(a) = \Lambda_B(b) = \lambda_{B_i}(b),$$

and so $j = i$. From this, it follows that

$$\lambda_{D_i}(a) = \lambda_{D_i}(b),$$

and so

$$\Lambda_D(a) = \lambda_{D_i}(a) = \lambda_{D_i}(b) = \Lambda_D(b).$$

If $j \in I \setminus \mathcal{B}$, then

$$\lambda_{B_i}(a) = \Lambda_B(a) = \Lambda_B(b) = (\lambda_{T_j}(b))\Phi_{j,i_B} \in T_{i_B},$$

and so $i = i_B \leq i_D$. This and the above $i_D \leq i$ together imply $i = i_B = i_D$. Then, by Lemma 1,

$$\Lambda_D(a) = \lambda_{D_i}(a) = \lambda_{B_i}(a).$$

As

$$\Lambda_D(b) = (\lambda_{T_j}(b))\Phi_{j,i_D},$$

we have

$$\Lambda_D(a) = \lambda_{B_i}(a) = (\lambda_{T_j}(b))\Phi_{j,i_B} = (\lambda_{T_j}(b))\Phi_{j,i_D} = \Lambda_D(b).$$

Case 2. $i \notin \mathcal{D}$, but $i \in \mathcal{B}$. If $j \in \mathcal{B}$, then

$$\lambda_{B_i}(a) = \Lambda_B(a) = \Lambda_B(b) = \lambda_{B_j}(b),$$

and so $j = i$. Then $\Lambda_D(a) = \Lambda_D(b)$ (see the first subcase of Case 1). If $j \notin \mathcal{B}$, then

$$\lambda_{B_i}(a) = \Lambda_B(a) = \Lambda_B(b) = (\lambda_{T_j}(b))\Phi_{j,i_B},$$

and so $i = i_B$. Thus, $\lambda_{B_i}(a) \in T_i$, and so (by Lemma 1)

$$\lambda_{B_i}(a) = \lambda_{T_i}(a).$$

If $i_D = i_B (= i)$, then $T_i \subseteq D_i \subseteq B_i$, and so (by Lemma 1) $\lambda_{B_i}(a) = \lambda_{D_i}(a) = \lambda_{T_i}(a)$. Thus,

$$\Lambda_D(a) = \lambda_{D_i}(a) = \lambda_{B_i}(a) = (\lambda_{T_j}(b))\Phi_{j,i_B} = (\lambda_{T_j}(b))\Phi_{j,i_D} = \Lambda_D(b).$$

If $i_D > i_B (= i)$, then $A_i \cap D = \emptyset$, and so

$$\Lambda_D(a) = (\lambda_{T_i}(a))\Phi_{i,i_D},$$

and

$$\Lambda_D(b) = (\lambda_{T_j}(b))\Phi_{j,i_D}.$$

As

$$\lambda_{T_i}(a) = \lambda_{B_i}(a) = (\lambda_{T_j}(b))\Phi_{j,i_B},$$

we have

$$\begin{aligned} \Lambda_D(b) &= (\lambda_{T_j}(b))\Phi_{j,i_D} = (\lambda_{T_j}(b))(\Phi_{j,i_B} \circ \Phi_{i_B,i_D}) = ((\lambda_{T_j}(b))\Phi_{j,i_B})\Phi_{i_B,i_D} \\ &= (\lambda_{T_i}(a))\Phi_{i_B,i_D} = (\lambda_{T_i}(a))\Phi_{i,i_D} = \Lambda_D(a). \end{aligned}$$

Case 3. $i \notin \mathcal{B}$. If $j \in \mathcal{B}$, then we can prove (as in the second subcases of Case 1 and Case 2) that $\Lambda_D(a) = \Lambda_D(b)$. Consider the case when $j \notin \mathcal{B}$. Then

$$(\lambda_{T_i}(a))\Phi_{i,i_B} = \Lambda_B(a) = \Lambda_B(b) = (\lambda_{T_j}(b))\Phi_{j,i_B}.$$

Hence,

$$\begin{aligned} \Lambda_D(a) &= (\lambda_{T_i}(a))\Phi_{i,i_D} = (\lambda_{T_i}(a))(\Phi_{i,i_B} \circ \Phi_{i_B,i_D}) = ((\lambda_{T_i}(a))\Phi_{i,i_B})\Phi_{i_B,i_D} \\ &= ((\lambda_{T_j}(b))\Phi_{j,i_B})\Phi_{i_B,i_D} = (\lambda_{T_j}(b))(\Phi_{j,i_B} \circ \Phi_{i_B,i_D}) = (\lambda_{T_j}(b))\Phi_{j,i_D} = \Lambda_D(b). \end{aligned}$$

In all cases, we have that $\Lambda_B(a) = \Lambda_B(b)$ implies $\Lambda_D(a) = \Lambda_D(b)$ for every $a, b \in A$. Consequently,

$$\text{Ker } \Lambda_B \subseteq \text{Ker } \Lambda_D.$$

Hence, A is a Boolean-type retractable automaton. \square

By Theorem 3, we can focus our attention on a Boolean-type retractable automaton containing a kernel. In our investigation two notions will play an important role. These notions are the dilation of automata and the semi-connected automata.

Definition 4. Let \mathbf{B} be an arbitrary subautomaton of an automaton $\mathbf{A} = (A, X, \delta)$. We say that \mathbf{A} is a dilation of \mathbf{B} if there exists a mapping $\phi_{\text{dil}}(\cdot)$ of A onto B that leaves the elements of B fixed, and fulfils $\delta(a, x) = \delta_B(\phi_{\text{dil}}(a), x)$ for all $a \in A$ and $x \in X$ ([5]).

If a is an arbitrary state of an automaton \mathbf{A} , then let $\mathbf{R}(a)$ denote the subautomaton generated by the element a (the smallest subautomaton containing a). It is easy to see that

$$R(a) = \{\delta(a, x) : x \in X^*\},$$

where X^* is the free monoid over X . Let us define the following relation:

$$\mathcal{R} := \{(a, b) \in A \times A : R(a) = R(b)\}.$$

It is evident that \mathcal{R} is an equivalence relation on A . The \mathcal{R} class containing a particular a element is denoted by R_a . The set $R(a) \setminus R_a$ is denoted by $R[a]$. It is clear that $R[a]$ is either an empty set or $\mathbf{R}[a]$ a subautomaton of \mathbf{A} . The factor automaton $\mathbf{R}\{a\} = \mathbf{R}(a)/\rho_{R[a]}$ is called a principal factor of \mathbf{A} . If $R[a]$ is an empty set, then consider $\mathbf{R}\{a\}$ as $\mathbf{R}(a)$ ([6]).

An \mathbf{A} automaton is said to be strongly connected if, for any $a, b \in A$, there exists a word $p \in X^+$ such that $\delta(a, p) = b$; (X^+ is the free semigroup over X).

Remark. For a word $p = x_1x_2 \dots x_n$ and an element a , the transition function is defined as the following:

$$\delta(a, p) = \delta(\dots \delta(\delta(a, x_1), x_2) \dots, x_n).$$

An automaton is called strongly trap-connected if it contains exactly one trap, and, for every $a \in A \setminus \{\text{trap}\}$ and $b \in A$, there is a word $p \in X^+$ such that $\delta(a, p) = b$.

An automaton is said to be semi-connected if its every principal factor is either strongly connected or strongly trap-connected ([6]).

Theorem 4 ([6]). *A state-finite automaton without outputs is a retractable automaton if and only if it is a dilation of a semi-connected retractable automaton.*

The next theorem is the extension of Theorem 4 to the Boolean-type retractable case.

Theorem 5. *A state-finite automaton without outputs is a Boolean-type retractable automaton if and only if it is a dilation of a semi-connected Boolean-type retractable automaton.*

PROOF. Let \mathbf{A} be a Boolean-type retractable state-finite automaton without outputs. Then, by Theorem 4, \mathbf{A} is a dilation of the retractable semi-connected automaton \mathbf{B} . For a subautomaton \mathbf{C} of \mathbf{B} , let λ'_C denote the restriction of λ_C to \mathbf{B} . It is easy to see that \mathbf{B} is a Boolean-type retractable automaton with the family $\{\lambda'_C \mid \mathbf{C} \text{ is a subautomaton of } \mathbf{B}\}$.

Conversely, let the automaton $\mathbf{A} = (A, X, \delta)$ be a dilation of the (sub)automaton $\mathbf{B} = (B, X, \delta_B)$. Let $\phi_{\text{dil}}(\cdot)$ denote the corresponding dilation of \mathbf{A} onto \mathbf{B} . Assume that \mathbf{B} is Boolean-type retractable (with a fixed family $\mathcal{F}_B = \{\lambda_C \mid \mathbf{C} \text{ is a subautomaton of } \mathbf{B}\}$ of retract homomorphisms of \mathbf{B}). Let \mathbf{C} be a subautomaton of \mathbf{A} . For every $c \in C$ and $x \in X$, $C \ni \delta(c, x) = \delta(\phi_{\text{dil}}(c), x) \in B$. Thus, $C \cap B \neq \emptyset$. Define the mapping Λ_C of A onto C as follows (see the proof of [6, Theorem 5]): if $a \in C$, then let $\Lambda_C(a) = a$. If $a \notin C$, then let $\Lambda_C(a) = \lambda_{C \cap B}(\phi_{\text{dil}}(a))$, where $\lambda_{C \cap B}$ is the element of \mathcal{F}_B corresponding to the subautomaton $\mathbf{C} \cap \mathbf{B}$ of \mathbf{B} . By the proof of [6, Theorem 5], Λ_C is a retract homomorphism of \mathbf{A} onto \mathbf{C} . We show that the family $\{\Lambda_C \mid \mathbf{C} \text{ is a subautomaton of } \mathbf{A}\}$ satisfies the conditions of Definition 2. Let $\mathbf{C}_1 \subseteq \mathbf{C}_2$ be arbitrary subautomata of \mathbf{A} . As \mathbf{B} is a Boolean-type retractable automaton, we have $\text{Ker } \lambda_{C_2 \cap B} \subseteq \text{Ker } \lambda_{C_1 \cap B}$. Assume $(a, b) \in \text{Ker } \Lambda_{C_2}$ for some $a, b \in A$ with $a \neq b$. Then $\Lambda_{C_2}(a) = \Lambda_{C_2}(b)$, and so $a, b \notin C_2$ or $a \in C_2, b \notin C_2$ or $a \notin C_2, b \in C_2$.

If $a, b \notin C_2$, then

$$\lambda_{C_2 \cap B}(\phi_{\text{dil}}(a)) = \Lambda_{C_2}(a) = \Lambda_{C_2}(b) = \lambda_{C_2 \cap B}(\phi_{\text{dil}}(b)),$$

and so

$$(\phi_{\text{dil}}(a), \phi_{\text{dil}}(b)) \in \text{Ker } \lambda_{C_2 \cap B} \subseteq \text{Ker } \lambda_{C_1 \cap B},$$

from which it follows that

$$\Lambda_{C_1}(a) = \lambda_{C_1 \cap B}(\phi_{\text{dil}}(a)) = \lambda_{C_1 \cap B}(\phi_{\text{dil}}(b)) = \Lambda_{C_1}(b),$$

because $a, b \notin C_1$. Thus, $(a, b) \in \text{Ker } \Lambda_{C_1}$.

If $a \in C_2, b \notin C_2$, then $\lambda_{C_2}(a) = a \in C_2$, and

$$a = \Lambda_{C_2}(a) = \Lambda_{C_2}(b) = \lambda_{C_2 \cap B}(\phi_{\text{dil}}(b)) \in B \cap C_2.$$

Thus,

$$\lambda_{C_2 \cap B}(a) = a = \lambda_{C_2 \cap B}(\phi_{\text{dil}}(b)),$$

and so $(a, \phi_{\text{dil}}(b)) \in \text{Ker } \lambda_{C_2 \cap B}$. Hence, $(a, \phi_{\text{dil}}(b)) \in \text{Ker } \lambda_{C_1 \cap B}$, that is,

$$\lambda_{C_1 \cap B}(a) = \lambda_{C_1 \cap B}(\phi_{\text{dil}}(b)) = \Lambda_{C_1}(b).$$

If $a \in C_1$, then $\lambda_{C_1 \cap B}(a) = a = \Lambda_{C_1}(a)$, and so

$$\Lambda_{C_1}(a) = \Lambda_{C_1}(b).$$

If $a \notin C_1$, then

$$\Lambda_{C_1}(a) = \lambda_{C_1 \cap B}(\phi_{\text{dil}}(a)) = \lambda_{C_1 \cap B}(a) = \Lambda_{C_1}(b).$$

In both subcases $(a, b) \in \text{Ker } \Lambda_{C_1}$.

Similar to the previous case, the assumption $a \notin C_2, b \in C_2$ implies $(a, b) \in \text{Ker } \Lambda_{C_1}$. In all three cases we have $(a, b) \in \text{Ker } \Lambda_{C_1}$. Hence, $\text{Ker } \Lambda_{C_2} \subseteq \text{Ker } \Lambda_{C_1}$. Thus, \mathbf{A} is a Boolean-type retractable automaton. \square

By Theorem 5 and Theorem 3, we can concentrate our attention on semi-connected automata containing kernels.

Definition 5. Let (T, \leq) be a partially ordered set, in which every two element subset has a lower bound, and every non-empty subset of T having an upper bound contains a maximal element. Consider the operation on T which maps a couple $(t_1, t_2) \in T \times T$ to the (unique) greatest upper bound of the set $\{t_1, t_2\}$. T is a semilattice under this operation. This semilattice is called a tree. It is clear that every finite tree has a least element ([7]).

If a non-trivial state-finite automaton \mathbf{A} contains exactly one trap a_0 , then A^0 will denote the set $A \setminus a_0$. If A is a trivial automaton, then let $A^0 = A$. On the set $A^0 \times X$ we consider a partial (transition) function δ^0 which is defined only on couples (a, x) for which $\delta(a, x) \in A^0$; in this case, $\delta^0(a, x) = \delta(a, x)$. We shall say that (A^0, X, δ^0) is the partial automaton derived from the automaton \mathbf{A} .

If \mathbf{A}^0 and \mathbf{B}^0 are partial automata, then a mapping ϕ of A^0 into B^0 is called a partial homomorphism of \mathbf{A}^0 into \mathbf{B}^0 if, for every $a \in A^0$ and $x \in X$, the condition $\delta_A(a, x) \in A^0$ implies $\delta_B(\phi(a), x) \in B^0$ and $\delta_B(\phi(a), x) = \phi(\delta(a, x))$.

Construction ([6]). Let (T, \leq) be a finite tree with the least element i_0 . Let $i \succ j$ ($i, j \in T$) denote the fact that $i \geq j$, and, for all $k \in T$, the condition $i \geq k \geq j$ implies $i = k$ or $j = k$. Let $\mathbf{A}_i = (A_i, X, \delta_i)$, $i \in T$ be a family of pairwise disjoint automata satisfying the following conditions:

- (i) \mathbf{A}_{i_0} is strongly connected, and \mathbf{A}_i is strongly trap-connected for every $i \in T, i \neq i_0$.

- (ii) Let $\phi_{i,i}$ denote the identical mapping of \mathbf{A}_i . Assume that, for every $i, j \in T, i \succ j$, there exists a homomorphism $\phi_{i,j}$ which maps \mathbf{A}_i^0 into \mathbf{A}_j^0 such that
- (iii) for every $i \succ j$ there exist elements $a \in A_i^0$ and $x \in X$ such that $\delta_i(a, x) \notin A_i^0$, $\delta_j(\phi_{i,j}(a), x) \in A_j^0$.

For arbitrary elements $i, j \in T$ with $i \geq j$, we define a partial homomorphism $\Phi_{i,j}(\cdot)$ of \mathbf{A}_i^0 into \mathbf{A}_j^0 as follows: $\Phi_{i,i} = \phi_{i,i}$, and, if $i \succ j$ such that $i \succ k_1 \succ \dots \succ k_n \succ j$, then let

$$\Phi_{i,j} = \phi_{k_n,j} \circ \phi_{k_{n-1},k_n} \circ \dots \circ \phi_{k_1,k_2} \circ \phi_{i,k_1}.$$

(We note that if $i \geq j \geq k$ are arbitrary elements of T , then $\Phi_{i,k} = \Phi_{j,k} \circ \Phi_{i,j}$.)

Let $A = \bigcup_{i \in T} A_i^0$. Define a transition function $\delta' : A \times X \mapsto A$ as follows. If $a \in A_i^0$ and $x \in X$, then let

$$\delta'(a, x) = \delta_{i'[a,x]}(\Phi_{i,i'[a,x]}(a), x),$$

where $i'[a, x]$ denotes the greatest element of the set $\{j \in T : \delta_j(\Phi_{i,j}(a), x) \in A_j^0\}$. It is clear that $\mathbf{A} = (A, X, \delta')$ is an automaton which will be denoted by $(A_i, X, \delta_i; \phi_{i,j}, T)$.

Theorem 6 ([6]). *A state-finite automaton without outputs is a semi-connected retractable automaton containing a kernel if and only if it is isomorphic to an automaton $(A_i, X, \delta_i; \phi_{i,j}, T)$ defined in the Construction.*

Remark 1. By the proof of [6, Theorem 7], if \mathbf{R} is a subautomaton of an automaton $(A_i, X, \delta_i; \phi_{i,j}, T)$ constructed as above, then there is an ideal $\Gamma \subseteq T$ such that $R = \bigcup_{j \in \Gamma} A_j^0$. As T is a tree,

$$\pi : i \mapsto \max\{\gamma \in \Gamma : \gamma \leq i\}$$

is a well-defined mapping of T onto Γ , which leaves the elements of Γ fixed. By the proof of [6, Theorem 7], λ_R defined by $\lambda_R(a) = \Phi_{i,\pi(i)}(a)$ ($a \in A_i^0$) is a retract homomorphism of \mathbf{A} onto \mathbf{R} . This fact will be used in the proof of the next Theorem.

Theorem 7. *A state-finite automaton without outputs is a semi-connected Boolean-type retractable automaton containing a kernel if and only if it is isomorphic to an automaton $(A_i, X, \delta_i; \phi_{i,j}, T)$ defined in the Construction.*

PROOF. Let \mathbf{A} be a state-finite automaton without outputs which contains a kernel. Assume that \mathbf{A} is also semi-connected and Boolean-type retractable. Then, by Theorem 6, \mathbf{A} is isomorphic to an automaton $\mathbf{A}=(A_i, X, \delta_i; \phi_{i,j}, T)$ which is defined in the Construction.

The main part of the proof is to show that every automaton $\mathbf{A}=(A_i, X, \delta_i; \phi_{i,j}, T)$ defined in the Construction is Boolean-type retractable. According to Theorem 6, the automaton $\mathbf{A}=(A_i, X, \delta_i; \phi_{i,j}, T)$ is retractable. Let \mathbf{B} be a subautomaton of \mathbf{A} . By Remark 1, there is an ideal $\Gamma_B \subseteq T$ such that $B = \cup_{i \in \Gamma_B} A_i^0$. Let $\pi_B(\cdot)$ be the mapping of T into itself, defined by $\pi_B : i \mapsto \max\{\gamma \in \Gamma_B; \gamma \leq i\}$. For an element $a \in A_i^0$ ($i \in T$), let $\lambda_B(a) = \Phi_{i, \pi_B(i)}(a)$. Using also Remark 1, it is easy to see that λ_B is a retract homomorphism of \mathbf{A} onto \mathbf{B} . Let \mathbf{B}_1 and \mathbf{B}_2 be arbitrary subautomata of $B_1 \subseteq B_2$ resp. We will show that $\text{Ker } \lambda_{B_2} \subseteq \text{Ker } \lambda_{B_1}$. Assume $\lambda_{B_2}(a) = \lambda_{B_2}(b)$, for some $a \in A_i^0$ and $b \in A_j^0$. Then $\Phi_{i, \pi_{B_2}(i)}(a) = \Phi_{j, \pi_{B_2}(j)}(b)$, from which it follows that $\pi_{B_2}(i) = \pi_{B_2}(j)$, and so $\pi_{B_1}(i) = \pi_{B_1}(j)$ because $B_1 \subseteq B_2$. Thus,

$$\begin{aligned} \lambda_{B_1}(a) &= \Phi_{i, \pi_{B_1}(i)}(a) = (\Phi_{\pi_{B_2}(i), \pi_{B_1}(i)} \circ \Phi_{i, \pi_{B_2}(i)})(a) = \Phi_{\pi_{B_2}(i), \pi_{B_1}(i)}(\lambda_{B_2}(a)) \\ &= \Phi_{\pi_{B_2}(i), \pi_{B_1}(i)}(\lambda_{B_2}(b)) = (\Phi_{\pi_{B_2}(i), \pi_{B_1}(i)} \circ \Phi_{j, \pi_{B_2}(j)})(b) \\ &= (\Phi_{\pi_{B_2}(i), \pi_{B_1}(i)} \circ \Phi_{j, \pi_{B_2}(i)})(a) = \Phi_{j, \pi_{B_1}(j)}(b) = \lambda_{B_1}(b). \end{aligned}$$

Consequently, $\text{Ker } \lambda_{B_2} \subseteq \text{Ker } \lambda_{B_1}$. Hence, $\mathbf{A}=(A_i, X, \delta_i; \phi_{i,j}, T)$ is a Boolean-type retractable automaton with the family $\{\lambda_B \mid \mathbf{B} \text{ is a subautomaton of } \mathbf{A}\}$. \square

ACKNOWLEDGEMENTS. I would like to thank ATTILA NAGY for his valuable remarks on the original version of the paper.

References

- [1] I. BABCSÁNYI and A. NAGY, Retractable Mealy-automata, *Pure Math. Appl.* **3** (1992), 147–153.
- [2] I. BABCSÁNYI and A. NAGY, Boolean-type retractable automata, *Publ. Math. Debrecen* **48** (1996), 193–200.
- [3] A. H. CLIFFORD and G. B. PRESTON, The Algebraic Theory of Semigroups I, *American Mathematical Society, Providence, RI*, 1961.
- [4] F. GÉCSEGE and I. PEÁK, Algebraic Theory of Automata, *Akadémiai Kiadó, Budapest*, 1972.
- [5] A. NAGY, Boolean-type retractable automata with traps, *Acta Cybernet.* **10** (1991), 53–64.
- [6] A. NAGY, Retractable state-finite automata without outputs, *Acta Cybernet.* **16** (2004), 399–409.

- [7] E. J. TULLY, Semigroups in which each ideal is a retract, *J. Austral Math. Soc.* **9** (1969), 239–245.

MÁRK FÜZESDI
DEPARTMENT OF ALGEBRA
BUDAPEST UNIVERSITY OF
TECHNOLOGY AND ECONOMICS
H-1521 BUDAPEST
P. O. BOX 91
HUNGARY

E-mail: fuzesdim@math.bme.hu

(Received October 1, 2015; revised June 30, 2016)