RAIRO-Theor. Inf. Appl. 45 (2011) 459–466
DOI: 10.1051/ita/2011128

Available online at:
www.rairo-ita.org

CLOSURE PROPERTIES OF HYPER-MINIMIZED AUTOMATA

Andrzej Szepietowski[1]

Abstract. Two deterministic finite automata are almost equivalent if they disagree in acceptance only for finitely many inputs. An automaton A is hyper-minimized if no automaton with fewer states is almost equivalent to A. A regular language L is canonical if the minimal automaton accepting L is hyper-minimized. The asymptotic state complexity $s^*(L)$ of a regular language L is the number of states of a hyper-minimized automaton for a language finitely different from L. In this paper we show that: (1) the class of canonical regular languages is not closed under: intersection, union, concatenation, Kleene closure, difference, symmetric difference, reversal, homomorphism, and inverse homomorphism; (2) for any regular languages L_1 and L_2 the asymptotic state complexity of their sum $L_1 \cup L_2$, intersection $L_1 \cap L_2$, difference $L_1 - L_2$, and symmetric difference $L_1 \oplus L_2$ can be bounded by $s^*(L_1) \cdot s^*(L_2)$. This bound is tight in binary case and in unary case can be met in infinitely many cases. (3) For any regular language L the asymptotic state complexity of its reversal L^R can be bounded by $2^{s^*(L)}$. This bound is tight in binary case. (4) The asymptotic state complexity of Kleene closure and concatenation cannot be bounded. Namely, for every $k \geq 3$, there exist languages K, L, and M such that $s^*(K) = s^*(L) = s^*(M) = 1$ and $s^*(K^*) = s^*(L \cdot M) = k$. These are answers to open problems formulated by Badr *et al.* [*RAIRO-Theor. Inf. Appl.* **43** (2009) 69–94].

Mathematics Subject Classification. 68Q45, 68Q70.

Keywords and phrases. Finite state automata, regular languages, hyper-minimized automata.

[1] Institute of Informatics, University of Gdańsk, ul. Wita Stwosza 57, 80-952 Gdańsk, Poland. matszp@inf.ug.edu.pl

Article published by EDP Sciences

1. Introduction

Badr *et al.* in [1] presented a polynomial-time algorithm for reducing a given deterministic finite automaton (dfa) into hyper-minimized dfa. The resulting automaton may disagree with the given dfa only on finitely many inputs, and is the smallest among such almost equivalent automata. The algorithm has been improved in [3].

The authors of [1] call a regular language canonical if its minimal dfa is hyper-minimized, and by asymptotic state complexity they mean the size of a hyper-minimized dfa for the language. They also formulated, among others, two open problems:

(1) what are the closure properties of canonical regular languages? We only know that this family forms a proper subset in the class of all regular languages and that it is closed under complement;

(2) similarly, we know next to nothing about asymptotic state complexity. For example, having given two regular languages, what can be told about the asymptotic state complexity of their intersection?

In this paper, we show that the class of canonical regular languages is not closed under intersection, union, concatenation, difference, symmetric difference, reversal, Kleene closure, homomorphism, and inverse homomorphism. The second part of the paper studies the asymptotic state complexity of basic regular operations. We provide upper bound mn for boolean operations, and 2^n for reversal, and show that both bounds are tight in the binary case. In the unary case, the upper bound on boolean operations can be met in infinitely many cases. The tight bound for reversal in the unary case is n since the reversal of every unary language is the same language. On the other hand, we prove that the asymptotic state complexity of Kleene closure and concatenation cannot be bounded.

2. Preliminaries

Two languages K and L are almost equivalent if they differ in finitely many elements, or in other words if their symmetric difference $K \oplus L = (K - L) \cup (L - K)$ is finite. Two deterministic finite automata A and B are almost equivalent if they disagree in acceptance only for finitely many inputs. An automaton A is hyper-minimized if no automaton with fewer states is almost equivalent to A. A regular language L is canonical if the minimal automaton accepting L is hyper-minimized. The asymptotic state complexity of a regular language L, denoted by $s^*(L)$, is the number of states of a hyper-minimized automaton for a language finitely different from L. Two states q_A and q_B are equivalent, if for each $w \in \Sigma^*$ we have $\delta(q_A, w) \in F$ if and only if $\delta(q_B, w) \in F$. States q_A and q_B are almost-equivalent if there exists $k \geq 0$ such that, for each $w \in \Sigma^*$ of length $|w| \geq k$, we have $\delta(q_A, w) \in F$ if and only if $\delta(q_B, w) \in F$. A state q is unreachable if it cannot be reached from the initial state by any input string. The state q is in the preamble, if it is reachable from the initial state, but only by finitely many inputs. Two natural numbers p and q are coprime if they have no common positive divisor other than 1. Note that for every natural $p > 1$, the numbers p and $p+1$ are coprime.

Lemma 2.1 (see [1], Thm. 3.4)**.** *A deterministic finite automaton A is hyper-minimized if and only if in A:*

(*a*) *there does not exists an unreachable state*;
(*b*) *there does not exists a pair of equivalent states*; *and*
(*c*) *there does not exists a pair of almost-equivalent states, such that at least one of them is in the preamble.*

Lemma 2.2 (see [1], Cor. 3.11)**.** *Let M be a minimal automaton with a cycle beginning and ending in its initial state. Then M is hyper-minimized.*

Lemma 2.3 (see [1])**.** *The class of canonical languages is closed under the complement, i.e. for every canonical language $L \subset \Sigma^*$, its complement $L^c = \Sigma^* - L$ is also canonical. Moreover, $s^*(L^c) = s^*(L)$.*

Lemma 2.4 (Chinese remainder theorem, see [2])**.** *Let p and q are two coprime numbers. Then for every two natural numbers x and y, there exists a natural number z such that*

$$z = x \pmod{p} \quad \textit{and}$$
$$z = y \pmod{q}.$$

3. Canonical regular languages

In this section we consider closure properties of canonical languages. Badr *et al.* [1] noted that they are closed under complement. We show that they are not closed under many other basic operations.

Lemma 3.1. *The class of canonical regular languages is not closed under*:

(*a*) *intersection*;
(*b*) *union*;
(*c*) *concatenation*;
(*d*) *Kleene closure*;
(*e*) *difference*;
(*f*) *symmetric difference*;
(*g*) *reversal*;
(*h*) *homomorphism*; *and*
(*i*) *inverse homomorphism.*

Proof.

(a) Consider two languages $L_a = a^*$ and $L_b = b^*$ over the alphabet $\{a, b\}$. The minimal automaton A_a accepting L_a has two states and the initial state is in a loop, so by Lemma 2.2, it is hyper-minimized and the language L_a is canonical. Similarly, L_b is canonical. The intersection $L_a \cap L_b = \{\lambda\}$ is not canonical over the alphabet $\{a, b\}$. The minimal automaton accepting $\{\lambda\}$ has two states and is almost-equivalent to the automaton with one state accepting the empty set $\emptyset$;

(b) follows from (a) and Lemma 2.3;

(c) consider two languages $L_2 = (aa)^*$ and $L_3 = (aaa)^*$ over the alphabet $\{a\}$. The minimal automaton accepting L_2 consists of the loop of the length 2 and the minimal automaton accepting L_3 consists of the loop of length 3. By Lemma 2.2, they are both hyper-minimized, hence L_2 and L_3 are canonical. The concatenation $L_2 \cdot L_3 = a^* - \{a\}$ is not canonical. The minimal automaton accepting $L_2 \cdot L_3$ has three states and is almost-equivalent to the automaton with one state accepting a^*;

(d) the empty set over the alphabet $\{a\}$ is canonical (its minimal automaton uses just one state), but its Kleene closure $\emptyset^* = \{\lambda\}$ is not canonical over the alphabet $\{a\}$. The corresponding automaton needs two states and is almost equivalent to the automaton with one state accepting the empty set. One can also consider the language $a^2(a^3)^*$ which is canonical, but its Kleene closure $a^* - \{a, aaa\}$ is not;

(e) and (f) Consider two canonical languages $L = bba(aa)^* + b$ and $K = bba(aa)^*$ over the alphabet $\{a, b\}$. They both are accepted by the minimal automata with five states and it is easy to check that the automata are hyper-minimized. The difference $L - K$ and the symmetric difference $L \oplus K$ are both equal to the non canonical language $\{b\}$;

(g) consider once again the canonical language $L = bba(aa)^* + b$. Its reversal $L^R = a(aa)^*bb + b$ is not canonical. The minimal automaton accepting L^R has six states and is almost-equivalent to the minimal automaton with five states accepting $a(aa)^*bb$;

(h) consider the canonical language $L_a^c = a^*b(a + b)^*$ – the complement of the canonical language $L_a = a^*$ over the alphabet $\{a, b\}$, and the homomorphism $h : (a+b)^* \to (a+b)^*$ which maps both a and b into a. The image $h(L_a^c) = a^+$ is not canonical over $\{a, b\}$. Indeed, it is accepted by the minimal automaton with three states and is almost-equivalent to the language a^* accepted with two states;

(i) consider the canonical language $L_b = b^*$ over the alphabet $\{a, b\}$ and the same homomorphism h as in (h). The inverse image $h^{-1}(L_b) = \{\lambda\}$ is not canonical over the alphabet $\{a, b\}$. □

4. Asymptotic state complexity

In this section we study the asymptotic state complexity of basic regular operations. We provide upper the bound mn for boolean operations, and 2^n for reversal, and show that the both bounds are tight in the binary case. In the unary case, the upper bound on boolean operations can be met in infinitely many cases. The tight bound for reversal in the unary case is n since the reversal of every unary language is the same language. On the other hand, we prove that the asymptotic state complexity of Kleene closure and concatenation cannot be bounded.

Lemma 4.1.

(a) *For any regular languages L_1 and L_2, the asymptotic state complexity of their sum $L_1 \cup L_2$, intersection $L_1 \cap L_2$, difference $L_1 - L_2$, and symmetric difference $L_1 \oplus L_2$ are all not greater than $s^*(L_1) \cdot s^*(L_2)$;*

(b) *for any two natural numbers $m, n > 1$, there exist languages L_m and L_n such that $s^*(L_m) = m$, $s^*(L_n) = n$, and*

$$s^*(L_m \cap L_n) = s^*(L_m \cup L_n) = s^*(L_m - L_n) = s^*(L_m \oplus L_n) = m \cdot n.$$

Proof.
(a) Let $A_1 = (Q_1, \Sigma, \delta_1, q_{I,1}, F_1)$ and $A_2 = (Q_2, \Sigma, \delta_2, q_{I,2}, F_2)$ be hyper-minimized automata accepting languages almost equivalent to L_1 and L_2, respectively. We can construct their cross-product $A = (Q, \Sigma, \delta, q_I, F)$ with $Q = Q_1 \times Q_2$, $q_I = (q_{I,1}, q_{I,2})$, $F = F_1 \times F_2$ and δ defined by

$$\delta((q_1, q_2), a) = (\delta(q_1, a), \delta(q_2, a)).$$

The automaton A has $s^*(L_1) \cdot s^*(L_2)$ states and accepts the intersection $L(A_1) \cap L(A_2)$ which is almost equivalent to the language $L_1 \cap L_2$. Hence, there exists a hyper-minimized automaton accepting a language almost equivalent to $L_1 \cap L_2$ with at most $s^*(L_1) \cdot s^*(L_2)$ states. Hence, $s^*(L_1 \cap L_2) \leq s^*(L_1) \cdot s^*(L_2)$.

In order to prove the other cases one only has to change the set of accepting states. For example, if we change the set of accepting states to $F = F_1 \times Q_2 \cup Q_1 \times F_2$, then the automaton A accepts the union $L(A_1) \cup L(A_2)$ which is almost equivalent to $L_1 \cup L_2$.

(b) Consider two languages

$$L_m = ((b^*ab^*)^m)^* \qquad \text{and} \qquad L_n = ((a^*ba^*)^n)^*.$$

The minimal automaton A_m accepting L_m has m states, say $Q_m = \{0, \ldots, m-1\}$, 0 is the initial and the only accepting state, and the transition function is defined in the following way:

$$\delta(x, a) = x + 1 \pmod{m} \quad \text{and} \quad \delta(x, b) = x, \quad \text{for every state } x \in Q_m.$$

The automaton A_m counts a-s modulo m and is hyper-minimized, since there is a loop going through the initial state. Similarly we can construct the hyper-minimized automaton A_n with states $Q_n = \{0, \ldots, n-1\}$ which counts b-s and accepts L_n. Hence $s^*(L_m) = m$ and $s^*(L_n) = n$. The languages $L_m \cap L_n$, $L_m \cup L_n$, $L_m - L_n$, and $L_m \oplus L_n$ are all accepted by the cross-product of A_m and A_n with the set of states $Q = Q_m \times Q_n$ and the appropriate set of accepting states. We shall concentrate on the automaton accepting the union $L_m \cup L_n$ with the set of accepting state $F = \{(0, y) \mid y \in Q_n\} \cup \{(x, 0) \mid x \in Q_m\}$ (other cases can be proved similarly). There is a loop going through the initial state $(0, 0)$, so by Lemma 2.2, we only have to show that the automaton is minimal, *i.e.* that no two states p and q are equivalent. It is enough to consider the following three cases:

Case 1. Both states $p = (p_1, p_2)$ and $q = (q_1, q_2)$ are not accepting, with some $p_1, p_2, q_1, q_2 \neq 0$, $p_1 \neq q_1$. In this case the state $\delta(p, a^{m-p_1})$ is accepting and the state $\delta(q, a^{m-p_1})$ is not.

Case 2. Both states p and q are accepting with $p = (p_1, 0)$, $q = (q_1, 0)$, and $p_1 \neq q_1$. In this case the state $\delta(p, a^{m-p_1}b)$ is accepting and the state $\delta(q, a^{m-p_1}b)$ is not.

Case 3. Both states p and q are accepting with $p = (p_1, 0)$, $q = (0, q_2)$ and $p_1, q_2 > 0$. In this case the state $\delta(p, a)$ is accepting and the state $\delta(q, a)$ is not.

□

Lemma 4.2. *For any two coprime numbers $p, q > 2$, there exist unary languages L_p and L_q such that $s^*(L_p) = p$, $s^*(L_q) = q$, and*

$$s^*(L_p \cap L_q) = s^*(L_p \cup L_q) = s^*(L_p - L_q) = s^*(L_p \oplus L_q) = p \cdot q.$$

Proof. Consider two languages

$$L_p = (a^p)^* = \{a^n \mid n \quad \text{is divisible by} \quad p\},$$

$$L_q = (a^q)^* = \{a^n \mid n \quad \text{is divisible by} \quad q\}.$$

The minimal automaton accepting L_p consists of the loop of the length p and the minimal automaton accepting L_q consists of the loop of length q. By Lemma 2.2, they are both hyper-minimized, hence $s^*(L_p) = p$ and $s^*(L_q) = q$. The languages $L_p \cap L_q$, $L_p \cup L_q$, $L_p - L_q$, and $L_p \oplus L_q$ are all accepted by the automata with one big loop of the length $p \cdot q$ and with different set of accepting states. More precisely, consider the automaton A with the set of states $Q = \{0, \ldots, pq-1\}$, the initial state $q_0 = 0$, and the transition function defined by:

$$\delta(x, a) = x + 1 \pmod{pq}, \quad \text{for every state } x \in Q.$$

The set of accepting states is chosen in the following way:

- $F_1 = \{0\}$ for the intersection $L_p \cap L_q$;
- $F_2 = \{x \in Q \mid x \text{ is divisible by } p \text{ or by } q\}$ for the union $L_p \cup L_q$;

- $F_3 = \{x \neq 0 \mid x \text{ is divisible by } \ p\}$ for the difference $L_p - L_q$; and
- $F_4 = \{x \neq 0 \mid x \text{ is divisible by } \ p \text{ or by } \ q\}$ for the symmetric difference $L_p \oplus L_q$.

We shall show that in all these cases the automaton A is minimal and so, by Lemma 2.2, also hyper-minimized. It is easy to see that with F_1, every two different states are not equivalent. In the second case with $F = F_2$, let x and y are two different states. We may assume that $x \neq y \pmod{p}$. By Lemma 2.4, there exists a natural number r such that

$$r = pq - x \pmod{p} \quad \text{and}$$
$$r = pq - y + 1 \pmod{q}.$$

Then the state $\delta(x, a^r) = x + r \pmod{pq}$ is divisible by p and is accepting, and the state $\delta(y, a^r) = y + r \pmod{pq}$ is divisible neither by p nor by q and is not accepting. Hence x and y are not equivalent.

Similarly, one can show that the automaton A with the accepting set F_4 is minimal. If $x - y + 1 \neq 0 \pmod{q}$, then take r as in the previous case. If $x - y + 1 = 0 \pmod{q}$, then take r such that

$$r = pq - x \pmod{p} \quad \text{and}$$
$$r = pq - y + 2 \pmod{q}.$$

Similarly, one can show that the automaton A with the accepting set F_3 is minimal. □

Lemma 4.3.

(*a*) *For any regular language L,*

$$s^*(L^R) \leq 2^{s^*(L)};$$

(*b*) *furthermore, for every $k \geq 1$, there exists language L such that $s^*(L) = k$ and $s^*(L^R) = 2^k$.*

Proof.
(a) Let $A = (Q, \Sigma, \delta, q_I, F)$ be the hyper-minimized automaton accepting a language almost equivalent to L. The automaton $A_R = (Q_R, \Sigma, \delta_R, q_R, F_R)$ accepting a language almost equivalent to L^R and having $2^{|Q|}$ states can be defined in the following way:

- the set of states of Q_R is the power set 2^Q, consisting of all subsets of Q;
- the initial state $q_R = F$;
- the set of accepting states $F_R = \{S \subset Q \mid q_I \in S\}$;
- the transition function δ_R is defined by

$$\delta_R(S, \sigma) = \{s \in Q \mid \delta(s, \sigma) \in S\}.$$

(b) Jiraskova and Sebej in [4] show that, for every $k \geq 2$, there is a binary language L such that minimal automaton accepting L has k states and the minimal automaton accepting the reversal L^R has 2^k states. Both these automata are hyper-minimized. Hence $s^*(L) = k$ and $s^*(L^R) = 2^k$. □

Lemma 4.4. *For every $k \geq 3$, there exist languages K, L, and M, such that $s^*(K) = s^*(L) = s^*(M) = 1$ and $s^*(K^*) = s^*(L \cdot M) = k$.*

Proof. Consider the language $K = a^k$ over the alphabet $\{a\}$. It contains only one word and is almost equivalent to the empty language $\emptyset$, hence, $s^*(K) = 1$. The Kleene closure

$$K^* = (a^k)^* = \{a^n \mid n \quad \text{is divisible by} \quad k\}$$

is accepted by the hyper-minimized automaton with k states.

Consider the languages $L = (a+b)^{k-3}b$ and $M = (a+b)^*$ over the alphabet $\{a, b\}$. The language L is finite and is almost equivalent to the empty language $\emptyset$, the language M is accepted by the minimal automaton with one state, so $s^*(L) = s^*(M) = 1$. The concatenation

$$L \cdot M = (a+b)^{k-3}b(a+b)^*$$

is accepted by the hyper-minimized automaton with k states. □

Lemma 4.5. *For every $k \geq 1$, there exist two languages K and L over the unary alphabet $\{a\}$, such that $s^*(K) = s^*(L) = k$ and $s^*(K \cap L) = s^*(K \cup L) = s^*(K \cdot L) = s^*((L)^*) = 1$.*

Proof. Let $K = (a^k)^*$ and $L = K^c \cup \{\lambda\}$. In the proof of Lemma 4.2 we have shown that $s^*(K) = k$. The language L is almost equivalent to K^c, so $s^*(L) = s^*(K) = k$. On the other hand $K \cup L = a^*$, $K \cap L = \{\lambda\}$, $K \cdot L = a^*$, and $(L)^* = a^*$. Hence, $s^*(K \cap L) = s^*(K \cup L) = s^*(K \cdot L) = s^*((L)^*) = 1$. □

Acknowledgements. I would like to thank the referees for their helpful remarks on the paper, especially for drawing my attention to some open problems and to better versions of Lemmas 4.1b and 4.3b.

References

[1] A. Badr, V. Geffert and I. Shipman, Hyper-minimizing minimized deterministic finite state automata. *RAIRO-Theor. Inf. Appl.* **43** (2009) 69–94.

[2] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms*, 2nd edition. MIT Press and McGraw-Hill (2001).

[3] M. Holzer and A. Maletti, An n log n algorithm for hyper-minimizing a (minimized) deterministic automaton. *Theoret. Comput. Sci.* **411** (2010) 3404–3413.

[4] G. Jiraskova and J. Sebej, Note on reversal of binary regular languages, in *Proc. of DCFS 2011. Lect. Notes Comput. Sci.* **6808** (2011) 212–221.

Communicated by J. Hromkovic.
Received December 7, 2010. Accepted October 10, 2011.