Information Processing Letters 16 (1983) 147–151
North-Holland Publishing Company

15 April 1983

POLYNOMIAL COMPLETE PROBLEMS IN AUTOMATA THEORY

I.K. RYSTSOV

Institute of Cybernetics, Ukrainian Academy of Science, 252207 Kiev – 207, U.S.S.R.

Communicated by A. Ershov
Received 17 August 1982

Kozen (1977) proved that the emptiness problem for regular languages intersection is polynomial complete. In this paper we show that many other problems concerning deterministic finite state automata are polynomial complete and therefore intractable for solution. On the other hand, simplified versions of these problems can be solved in polynomial time by deterministic algorithms. This work is a part of the research on automata theory carried out at the Institute of Cybernetics headed by academician V.M. Glushkov.

Keywords: Automata theory, computational complexity, complete problems

1. Introduction

A *weakly-initial* automaton A is specified by five objects, A = (S, X, f, I, F), where S is a finite set of states, X a finite set of inputs, f a transition function which maps $S \times X$ into S, $I \subseteq S$ a nonempty set of initial states and $F \subseteq S$ a nonempty set of final states. A *non-initial* automaton is a weakly-initial automaton in which I = S. In [2] a non-initial automaton is called a multiple-entry automaton and is specified by four objects, A = (S, X, f, F). We denote the sets of weakly-initial and non-initial automata by WIA and NIA respectively. Of course NIA may be considered to be the subset of WIA.

X^* denotes the set of all input words. The transition function is extended by induction on $S \times X$ and its value is denoted by f(s, p) or simply sp for $s \in S$ and $p \in X^*$ [3]. The transition function is also extended on subsets of states by the formula

$$Tp = \{sp \mid s \in T\}$$

for $T \subseteq S$. The automaton $\mathfrak{A} = (B(S), X, \Gamma)$ where B(S) is the set of all subsets of S and $\Gamma(T, x) = Tx$ is called the global automaton for A [4]. The subautomaton of global automaton $\mathfrak{A}_2 = (B_2(S), X, \Gamma)$ defined on all subsets with at most two states is called a pair automaton for A [4].

The cardinality of a finite set V is denoted by |V|. The transition function of an automaton is described by a table with |X| rows and |S| columns. The number $|A| = |S| \cdot |X|$ is called the *size* of automaton A. The transition function and subset of states of an automaton are coded as a string of symbols in some finite alphabet. Thus, a weakly-initial automaton may be considered as a string of symbols, and $\langle A \rangle$ denotes the code of A.

In this paper the reader is assumed to be familiar with the definitions of the classes P, PSPACE and with the notion of polynomial complete problems [5]. We consider, for the sake of simplicity, only polynomial time reduction but all the results also hold for logarithmic space reduction.

2. Global inclusion problem

Let the following set of input words be associated with a weakly-initial automaton A:

$$I(A) = \{p \mid Ip \subseteq F\}.$$

In the case |I| = 1 this is the regular set of words accepted by an initial automaton [6]. The problem of whether I(A) is empty for given A is called the global inclusion problem or in short the inclusion problem. The inclusion problem for initial automata is solved by a deterministic algorithm in linear time [6]. This algorithm can be applied to global automaton to solve the global inclusion problem but this requires exponential time and space of the size of a given automaton.

Our futher aim is to prove polynomial completeness of the inclusion problem in class PSPACE which consists of the problems solved by deterministic algorithms in polynomial space. It is easy to see that the inclusion problem can be solved by a simple nondeterministic algorithm in linear space; therefore, from Savitch's theorem [5] it follows that the inclusion problem belongs to PSPACE. This reasoning can be applied to each problem considered in this paper; therefore, in sequel we shall omit the proofs of the problems belonging to PSPACE.

Theorem 2.1. *The global inclusion problem for non-initial automata is* PSPACE-*complete.*

Proof. We reduce Kozen's problem [1] to our problem. Let

$$A_i = (S_i, X, f_i, \{s_i\}, F_i), \quad 1 \leqslant i \leqslant m,$$

be an initial automata with common input alphabet and let L be an intersection of languages accepted by these automata. Without loss of generality, assume $S_i \cap S_j = \emptyset$ if $i \neq j$. Let S be a union of all states S_i, and F a union of all states F_i. Let $T = \{t_1, \ldots, t_m\}$ be the new states $t_i \notin S$ and let z be the new input symbol $z \notin X$.

Define non-initial automata

$$A = (S \cup T, X \cup \{z\}, f, F)$$

in the following way:

$$\begin{aligned} f(s, x) &= f_i(s, x) && \text{for all } s \in S_i \text{ and } x \in X,\\ f(s, z) &= s_i && \text{for all } s \in S_i,\ 1 \leqslant i \leqslant m,\\ f(t_i, x) &= t_i && \text{for all } x \in X \text{ and } 1 \leqslant i \leqslant m,\\ f(t_i, z) &= s_i && \text{for all } i,\ 1 \leqslant i \leqslant m. \end{aligned}$$

It is obvious that the automaton $\langle A \rangle$ can be constructed in linear time of the sum of sizes $|A_i|$, $1 \leqslant i \leqslant m$. If $p \in L$, then, from definition of A, it follows that $zp \in I(A)$. Conversely, if $q \in (X \cup z)^*$ is the shortest word in I(A), then it is clear that q must be identical to zp where $p \in L$. Thus $L \neq \emptyset$ iff $I(A) \neq \emptyset$ and the theorem is proved.

From this theorem it evidently follows that the inclusion problem for weakly-initial automata is also PSPACE-complete. Now we show that this theorem cannot be essentially improved. The input word p will be called *reset* for states s and t if $f(s, p) = f(t, p)$. Let k be a fixed natural number and WIA_k is the set of weakly-initial automata with $|F| \leqslant k$. The set NIA_k is defined analogously.

Theorem 2.2. *The inclusion problem for non-initial automata from* NIA_k *belongs to class* P.

Proof. We sketch the algorithm of solution. Let A be the given non-initial automaton with $|F| \leqslant k$ and let $N(A) = \{p \mid |Sp| \leqslant k\}$. It is easy to determine that $N(A) = \emptyset$ or to find some word $p \in N(A)$ in polynomial time of $|A|$ using reset words. We can determine the absence of a reset word for states s and t or we can find such a word in polynomial time using the pair automaton [4].

If $N(A) = \emptyset$, then $I(A) = \emptyset$. Otherwise, the subautomaton $(B_k(S), X, \Gamma)$ of global automaton is constructed on the subsets with at most k states in $O(|A|^k)$ time. Let B(F) be the set of all subsets of F and $p \in N(A)$. Consider the initial automaton

$$A_k = (B_k(S), X, \Gamma, \{Sp\}, B(F)).$$

It is evident that $I(A) = \emptyset$ iff $I(A_k) = \emptyset$ but the latter question can be solved in polynomial time of $|A_k|$.

Contrary to this theorem we shall shown in the next section that the inclusion problem for WIA_1 is PSPACE-complete.

3. Reachability and equivalence problems

The set of input words R(A) is associated with each weakly-initial automaton A = (S, X, f, I, F) as follows:

$R(A) = \{P \mid Ip = F\}$.

The problem of emptiness of R(A) for given A is called the global reachability problem or, in short, the reachability problem. The reachability and inclusion problems are equivalent in the case $|F| = 1$ because $|Ip| \geqslant 1$ for any p. Therefore, Theorem 2.2 implies the following.

Corollary 3.1. *The reachability problem for* NIA_1 *belongs to the class* P.

The following theorem shows that this bound cannot be improved if $P \neq PSPACE$.

Theorem 3.2. *The reachability problem for* NIA_2 *is* PSPACE-*complete.*

Proof. It suffices to reduce the inclusion problem to this problem in polynomial time and this theorem will be proved according to Theorem 2.1. Let $A = (S, X, f, F)$ be a given automaton. We shall add to it new states $t_1, t_2, t_3 \notin S$ and a new input symbol $z \notin X$ and consider a non-initial automaton

$$B = (S \cup \{t_1, t_2, t_3\}, X \cup \{z\}, g, \{t_1, t_3\})$$

where g is defined in the following way:

$$g(s, x) = f(s, x) \quad \text{for all } s \in S \text{ and } x \in X,$$
$$g(t_i, x) = g(t_i, z) = t_3 \quad \text{for all } x \in X,\ 1 \leqslant i \leqslant 3,$$
$$g(s, z) = \begin{cases} t_1 & \text{if } s \in F, \\ t_2 & \text{if } s \notin F. \end{cases}$$

It is obvious that automaton B can be constructed in linear time in $|A|$. Suppose that $p \in I(A)$; then from the construction of B it follows that $pz \in R(B)$. Conversely, if $q \in (X \cup z)^*$ is the shortest word in R(B), then q must be equal to pz and $p \in I(A)$. Thus $I(A) \neq \emptyset$ iff $R(B) \neq \emptyset$ and the theorem is proved.

Putting $I = S$ and $F = \{t_1\}$ in the automaton B described in this theorem we shall have the following.

Corollary 3.3. *The reachability and inclusion problems for* WIA_1 *are* PSPACE-*complete.*

We note without proof that the reachability and inclusion problems are PSPACE-complete for partial non-initial automata with $|F| = 1$.

Let the following set of input words L(A) also be associated with each automaton:

$L(A) = \{p \mid Ip \cap F \neq \emptyset\}$.

The problem whether $L(A) = X^*$ for given A is called the *complement emptiness problem.* Two automata A and B are called equivalent if $L(A) = L(B)$. The determination of equivalence of two given automata is called the *equivalence problem.*

Theorem 3.4. *The complement emptiness problem for non-initial automata is* PSPACE-*complete.*

Proof. Let $A = (S, X, f, F)$ be the given automaton; then consider the automaton $B = (S, X, f, S \setminus F)$. From the definitions it is easy to see that $L(B) = X^* \setminus I(A)$. Therefore, $I(A) = \emptyset$ iff $L(B) = X^*$ and the theorem is proved.

Corollary 3.5. *The equivalence problem for non-initial automata is* PSPACE-*complete.*

This corollary directly follows from Theorem 3.4 because the complement emptiness problem may be treated as the equivalence problem with one state automaton.

4. Automata with outputs

An automaton $\mathfrak{A}$ with outputs is defined by five objects,

$$\mathfrak{A} = (S, X, Y, \delta, \lambda)$$

where S, X and δ are the same as in weakly-initial automata, Y is a finite set of outputs, and λ an output function which maps $S \times X$ into Y. As usually λ is extended to $S \times X^*$ and the output word is denoted by $\lambda(s, p)$ [7]. Recall that two states are called equivalent if they produce identical output words for every input word.

The following set of experiments is associated with each $\mathfrak{A}$:

$$E(\mathfrak{A}) = \{(p, q) \mid (\exists s)\ \lambda(s, p) = q\}.$$

We call two automata $\mathfrak{A}_1$ and $\mathfrak{A}_2$ weakly-equivalent if $E(\mathfrak{A}_1) = E(\mathfrak{A}_2)$. Moore [8] proved that in two weakly equivalent strongly-connected automata for each state in one automaton there is an equivalent state in the other automaton. Therefore, the weak equivalence problem is solved in polynomial time in this case [7]. Now we shall see that in general this problem is intractable.

Theorem 4.1. *The weak equivalence problem is* PSPACE-*complete.*

Proof. Let $A_i = (S_i, X, f_i, F_i)$, $i = 1, 2$, be two given non-initial automata with common input alphabet. Let us consider two automata with outputs

$$\mathfrak{A}_i = (S_i \cup \{t_i\}, X \cup \{z\}, Y, \delta_i, \lambda_i),$$

where $t_i \notin S_i$, $z \notin X$ and $Y = \{0, 1\}$, and the transition functions are defined as follows:

$$\delta_i(s, x) = f_i(s, x) \quad \text{for all } s \in S_i,\ x \in X,\ i = 1, 2,$$

and $\delta_i(\cdot, \cdot) = t_i$ on all other states and inputs, $i = 1, 2$. Define an output function $\lambda_i(s, z) = 1$ for $s \in F_i$ and $\lambda_i(\cdot, \cdot) = 0$ on all other states and inputs, $i = 1, 2$.

Obviously, an automaton with outputs $\mathfrak{A}_i$ can be constructed in linear time of the size of a given automaton A_i. It is easy to verify that

$$L(A_1) = L(A_2) \quad \text{iff} \quad E(\mathfrak{A}_1) = E(\mathfrak{A}_2)$$

and therefore, according to Corollary 3.5, the theorem is proved.

Input word p is called diagnostic for the automaton $\mathfrak{A}$ if $\lambda(s, p) \neq \lambda(t, p)$ for all different states s and t [7]. Denote by $D(\mathfrak{A})$ the set of diagnostic words for $\mathfrak{A}$ and call the problem of emptiness of $D(\mathfrak{A})$ the *diagnosing problem.*

Theorem 4.2. *The diagnosing problem is* PSPACE-*complete.*

Proof. Let $A = (S, X, f, F)$ be a non-initial automaton and let $S = \{s_1, s_2, \ldots, s_n\}$. Define the automaton with outputs

$$\mathfrak{A} = (\{1, 2, \ldots, 2n\}, X \cup \{z\}, Y, \delta, \lambda)$$

where

$$z \notin X, \qquad Y = \{y_0, y_1, \ldots, y_n\}$$

as follows:

$$\delta(i, x) = j, \quad \delta(i + n, x) = j + n \qquad \text{if } f(s_i, x) = s_j,$$

and

$$\delta(i, z) = \delta(i + n, z) = 1 \quad \text{for all } i,\ 1 \leqslant i \leqslant n.$$

Moreover,

$$\lambda(i, x) = \lambda(i + n, x) = y_i \quad \text{for all } x \text{ and } i,$$

$$\lambda(i, z) = y_0 \quad \text{if } s_i \in F$$

and

$$\lambda(i, z) = y_1 \quad \text{on all other states } i,\ 1 \leqslant i \leqslant 2n.$$

It is evident that the automaton $\mathfrak{A}$ can be constructed in linear time in $|A|$. If p is a nonempty word in I(A), then from construction of $\mathfrak{A}$ it follows that $pz \in D(\mathfrak{A})$. Conversely, if $q \in (X \cup z)^*$ is the shortest word in $D(\mathfrak{A})$, then q must be equal to pz where $p \in I(A)$. Thus $I(A) \neq \emptyset$ iff $D(\mathfrak{A}) \neq \emptyset$ and the theorem is proved.

Finally we shall consider the homogeneous problem. Recall that the word p is called *homing* for the automaton $\mathfrak{A}$ if

$$\lambda(s, p) \neq \lambda(t, p) \quad \text{or} \quad \delta(s, p) = \delta(t, p) \quad \text{for all } s, t \in S.$$

An automaton with outputs is called a reduced automaton if there are no two equivalent states in it. In [8] Moore proved the existence of homing words for every reduced automaton but as far as the author knows the following simple result is not known.

Theorem 4.3. *A homing word exists for the automaton if and only if there is a reset word for every pair of equivalent states.*

Proof. *Necessity* is evident. *Sufficiency* is proved by simple modification of the method which was described in Theorem 2.2.

Corollary 4.4. *The homogeneous problem for automata with outputs belongs to class* P.

5. Conclusion

New polynomial complete problems concerning finite automata are presented in this paper. Some of these problems such as diagnostic and weak equivalence problems have been investigated for a long time by many authors, and great efforts have been expended in order to find the effective solution. It is interesting to note that in these problems there is no obvious quantifier alternation such as in well-known PSPACE-complete problems from logic and game theory [9].

Acknowledgment

The author is grateful to J.V. Kapitonova and A.A. Letichevski and A.N. Chebotarev for helpful discussions.

References

[1] D. Kozen, Lower bounds for natural proof systems, Proc. 18th Ann. Symp. on Foundations of Computer Science (1977) 254–266.
[2] A. Gill and L. Kou, Multiple-entry finite automata, J. Comput. Syst. Sci. 9 (1974) 1–19.
[3] V.M. Glushkov, Abstractnaja teoria avtomatov, Uspehi Mat. Nauk 16(5) (1961) 1–70 (in Russian).
[4] F.C. Hennie, Finite-State Models for Logical Machines (Wiley, New York, 1968).
[5] A.V. Aho, J.E. Hopcroft and J.D. Ullman, The Design and Analysis of Computer Algorithms (Addison-Wesley, Reading, MA, 1974).
[6] M.O. Rabin and D. Scott, Finite automata and their decision problems, IBM J. Res. Develop. 3 (1959) 114–125.
[7] A. Gill, Introduction to the Theory of Finite-State Machines (McGraw-Hill, New York, 1962).
[8] E.F. Moore, Gedanken experiments on sequential machines, in: Automata Studies (Princeton University Press, Princeton, 1956).
[9] M.R. Garey and D.S. Johnson, Computers and Intractability (Freeman, San Francisco, CA, 1979).