Theoretical Computer Science 412 (2011) 5808–5813

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Note

On the complexity of computing the profinite closure of a rational language

P.-C. Héam *

Projet INRIA-CASSIS, Laboratoire d'Informatique de l'Université de Franche-Comté, 16 route de Gray, 25030 Besançon Cedex, France

ARTICLE INFO

Article history:
Received 14 June 2007
Received in revised form 14 June 2011
Accepted 22 June 2011
Communicated by D. Perrin

Keywords:
Finite automata
State complexity
Profinite topology

ABSTRACT

Profinite topology is used in the classification of rational languages. In particular, several important decidability problems, related to the Malcev product, reduce to the computation of the closure of a rational language in the profinite topology. It is known that, given a rational language by a deterministic automaton, computing a deterministic automaton accepting its profinite closure can be done with an exponential upper bound. This paper is dedicated the study of a lower bound for this problem: we prove that, in some cases, if the alphabet contains at least three letters, it requires an exponential time.

1. Preliminaries

For more information on automata and language theory we refer the reader to [1,4,9]. For a general reference on profinite topologies, see [5,13].

1.1. Introduction

Profinite topology is used to characterize certain classes of rational languages: the languages of level 1/2 in the group hierarchy and the languages recognizable by reversible automata [11,14]. Moreover, profinite topologies on the free group or on the free monoid play a crucial role in the theory of finite semigroups [2,7,12,3]. In particular, several important decidability problems, related to the Malcev product, reduce to the computation of the closure of a rational language in the profinite topology.

It is known that the profinite closure of a rational language is rational too [16,8]. The first algorithm was given in [15] for languages given by rational expressions, while [17,10] provide algorithms on finite automata. In this paper we are interested in the following problem.

Profinite Closure
Input: A finite deterministic n-state automaton $\mathcal{B}$ on the alphabet A.
Output: A finite deterministic automaton accepting the profinite closure of $L(\mathcal{B})$.

A solution to this problem is known to be computable in time $O(2^n)$ [10]. We prove in this paper that it cannot be done, in some cases, faster than in exponential time (if the alphabet contains at least three letters).

In the first part of this paper, we introduce useful notation and definitions. In the second part, we recall an algorithm [17,10] to solve the above problem. The last section of this paper is dedicated to the main result of the paper: we will prove that there exists a family or rational languages K_n such that the following hold.

* Tel.: +33 3 81 66 66 53.
E-mail address: heampc@lifc.univ-fcomte.fr.

0304-3975/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2011.06.022

(1) The minimal automaton of K_n has $3n$ states,
(2) The minimal automaton of the profinite closure of K_n has $\Omega(4^n/\sqrt{n})$ states.

Notice that the topological notions related to this paper are technical and require a wide mathematical background. However, the proved result can be easily understood using only automata theoretic arguments. In order to not overload the reader, we do not present the mathematical background in this article. The interested reader is referred to [5,13] for more information on profinite topologies. Particular topological properties of rational languages are studied in [6]. We simply provide short definitions in the next section.

1.2. Background and notation

Let A be a finite alphabet, and let $\overline{A} = \{\overline{a} \mid a \in A\}$ be a copy of A. Finally, let $\tilde{A}$ be the disjoint union of A and $\overline{A}$. The map $a \mapsto \overline{a}$ from A onto $\overline{A}$ can be extended to a one-to-one function from $\tilde{A}$ into itself by setting $\overline{\overline{a}} = a$. A word of A^* is said to be reduced if it does not contain any factor of the form $a\overline{a}$ with $a \in \tilde{A}$. We denote by $\equiv$ the monoid congruence generated by the relations $a\overline{a} \equiv 1$ for all $a \in \tilde{A}$. The set $\tilde{A}/_{\equiv}$ is a group for the quotient law, called the free group over A. Let π be the projection from $\tilde{A}$ into this group, which is a monoid morphism. We denote by $D(\tilde{A})$ the set $\pi^{-1}(\varepsilon)$, i.e., the set of words of $\tilde{A}^*$ that can be rewritten into ε using the rewriting rules $a\overline{a} \to \varepsilon$, with $a \in \tilde{A}$.

The family of normal subgroups of the free group with finite index forms a basis of open sets for the profinite topology on the free group. Similarly, the class of group languages (regular languages whose syntactic monoids are finite groups) forms a basis of open sets for the profinite topology on A^*. Note that the profinite closure in A^* of a language L is the intersection of A^* with its profinite closure in the free group. Throughout this paper, the profinite topology considered is the one on A^*.

Recall that a *finite automaton* is a 5-tuple $\mathcal{A} = (Q, B, E, I, F)$, where Q is a finite set of *states*, B is the alphabet, $E \subseteq Q \times B \times Q$ is the set of *edges* (or *transitions*), $I \subseteq Q$ is the set of *initial states*, and $F \subseteq Q$ is the set of *final states*. A *path* in $\mathcal{A}$ is a finite sequence of consecutive edges:

$$p = (q_0, a_0, q_1), (q_1, a_1, q_2), \ldots, (q_{n-1}, a_n, q_n).$$

The *label* of the path p is the word $a_1a_2 \cdots a_n$, its *origin* is q_0, and its *end* is q_n. A word is accepted by $\mathcal{A}$ if it is the label of a path in $\mathcal{A}$ having its origin in I and its end in F. Such a path is said to be *successful*. The set of words accepted by $\mathcal{A}$ is denoted by $L(\mathcal{A})$.

For every state q and language K, we denote by $q \cdot_{\mathcal{A}} K$ (or $q \cdot K$ if there is no ambiguity on $\mathcal{A}$) the subset of Q of all the states which are the end of a path having its origin in q and its label in K. An automaton is said to be *trim* if for each state q there exists a path from an initial state to q and a path from q to a final state. An automaton is *deterministic* if it has a unique initial state and does not contain any pair of edges of the form (q, a, q_1) and (q, a, q_2) with $q_1 \neq q_2$. An important result of automata theory states that for any automaton $\mathcal{A}$ there exists exactly one deterministic automaton (up to isomorphism) with a minimal number of states which accepts the same language. It is called the *minimal automaton* of $L(\mathcal{A})$. Two states p and q of an automaton are *Nerode equivalent* if, for every word u, $p \cdot u$ is final if and only if $q \cdot u$ is final too. It is well known that a trim deterministic automaton is minimal if and only if all classes of the Nerode equivalence are singletons.

Let $\mathcal{A}$ be an automaton with set of states Q and set of transitions E. A subset P of Q is said to be *strongly connected* if, for each pair p and q of states in P, there exist a path from p to q and a path from q to p. A strongly connected component of $\mathcal{A}$ is a maximal (for the inclusion) set of states which is strongly connected. The strongly connected components of $\mathcal{A}$ form a partition of Q. A transition (p, a, q) of $\mathcal{A}$ is *internal to a strongly connected component* if p and q belong to the same strongly connected component. It is said to be *internal* if it is internal to some strongly connected component and *external* otherwise.

The class of *rational* languages of A^* is the smallest class of languages closed under product, finite union, and star operations. It is well known that a language of A^* is rational if and only if it can be accepted by a finite automaton.

1.3. Profinite closure of a rational language

It is known that the profinite closure of a rational language is rational too [16].

In this direction, we use the following algorithm [17,10], called ProfiniteClosure, working on a finite trim automaton $\mathcal{A} = (Q, A, E, I, F)$ in order to compute the profinite closure of $L(\mathcal{A})$.

1. Compute the strongly connected components of $\mathcal{A}$.
2. Compute the set T of external transitions.
3. Compute $E_1 = \{(q, a, p) \mid (p, \overline{a}, q) \in E \setminus T\}$. Let $\mathcal{A}_1 = (Q, \tilde{A}, E \cup E_1, I, F)$.
4. Compute $E_2 = \{(p, \varepsilon, q) \mid p \neq q,\ q \in p \cdot_{\mathcal{A}_1} u,\ u \in D(\tilde{A})\}$. Let $\mathcal{A}_2 = (Q, A, E \cup E_2, I, F)$.
5. Return $\mathcal{A}_3$, the automaton obtained for $(Q, A, E \cup E_2, I, F)$ by a classical ε-transition elimination.

In order to obtain a resulting deterministic automaton, one can use the standard determinization algorithm, which is known to be exponential in the worst case [9]. We illustrate how this algorithm works on a graphical example in Fig. 1.