数理解析研究所講究録 1437 巻 2005 年 11-16

SYNTACTIC MONOIDS AND LANGUAGES*

TERUYUKI MITOMA AND KUNITAKA SHOJI
DEPARTMENT OF MATHEMATICS, SHIMANE UNIVERSITY
MATSUE, SHIMANE, 690-8504 JAPAN

In this paper, we investigate the structures of syntactic monoids of languages and take up the related problems.

1 Syntactic monoids

Definition 1. X is finite alphabet, X^* is the set of words over X, L is a subset of X^*, is called a *language*. The *syntactic congruence* σ_L on X^* is defined by $w\sigma_L w'$ if and only if the sets $\{(x,y) \in X^* \times X^* \mid xwy \in L\}$, $\{(x,y) \in X^* \times X^* \mid xw'y \in L\}$ are equal to each other. The *syntactic monoid* of L is defined to be a monoid X^*/σ_L.

Definition 2. An finite *automaton* $\mathcal{A}$ is a quintuple

$$\mathcal{A} = (A, V, E, I, T)$$

where X is a finite alphabet, V is a finite set of states, E is a finite set of directed edges each of which is labelled by a letter of X ; edges e are written as $e = (v, a, v')$, where $v, v' \in V$ and $a \in X$. I is a subset of V , each of which is called an *initial* state, and T is a subset of V, each of which is called a *terminal* state.

Let L be a language over X. Then we say that L is a *regular* language over X if there exists an automaton $\mathcal{A}$ with $L = L(\mathcal{A})$.

Result 1. *Let L be a language over X. Then L is regular if and only if $Syn(L)$ is a finite monoid.*

Problem 1. Given a language L, discribe structure of $Syn(L)$.

Result 2. *Let L be a language of X^* and L^c the complement of the set L in X^*. Then $Syn(L) = Syn(L^c)$.*

Example 1. *Let $A = \{a_1, \cdots, a_n\}$.* Let L be a language of A^*. If the syntactic monoid $Syn(L)$ is a right zero semigroup with 1, then $Syn(L)$ is three-element semigroup.

---

*This is an absrtact and the paper will appear elsewhere.

Example 2. $A = \{a_1, \cdots, a_n\}$. For any $w = b_1 b_2 \cdots b_r$, let $w^R = b_r \cdots b_2 b_1$. Let $L = \{ww^R | w \in A^*\}$. Then $Syn(L)$ is the free monoid A^* on A.

Example 3. Let $A = \{a, b\}$ and $L = \{a^n b^n | n \in N\}$. Then all of σ_L-classes are $\{1\}$, $\{ab\}$, $\{a^n\}$, $\{b^n\}$, $c_n = \{a^{p+n} b^p | p \in N\}$, $d_n = \{a^q b^{q+n} | q \in N\}$, $0 = A^* ba A^*$. Also, $Syn(L) - \{0, 1\}$ is a $\mathcal{D}$-class.

Example 4 Let $A = \{a_1, \cdots, a_n\}$. Give the length and lexicographic ordering on A^* with $a_1 < \cdots < a_n$. Let w_n be the word obtained by juxtapointing words of length n to x_1^n from lower to upper in the the length and lexicographic ordering. For instance, $w_1 = a_1 \cdots a_n$,

$$w_2 = (a_1 a_1)(a_1 a_2) \cdots\cdots (a_1 a_n) \cdots (a_n a_{n-1})(a_n a_n) \text{ and so on.}$$

and let $L = \{w_n | n \in N\}$ be the set of words. The free monoid A^* on A is isomorphic to $Syn(L)$.

Example 5. Let $A = \{a_1, \cdots, a_r\}$ and let L be the set of words w_n in which each a_i occurs exactly n times. Then the free commutative monoid on A is isomorphic to $Syn(L)$.

Result 3. *For every finitely generated group G, there exists a language L of X^* such that G is isomorphic to $Syn(L)$.*

2 A-Graphs, Automata, and embedding of monoids in Syntactic monoids

Definition 3. Let A be a finite set. Then $G = (A, V, E)$ is a (*directed*) *A-graph*, where V is a set of vertices, E is a set of directed edges with a letter as label and so edges e from a vertex v to a vertex v' are written as $e = (v, a, v')$ or $e : v \xrightarrow{a} v'$.

A A-graph $G = (A, V, E)$ is said to be *deterministic* if $\forall v \in V$, $\forall a \in A$, there exists at most one vertex $v' \in V$ such that $(v, a, v') \in E$.

Assume that a A-graph $G = (A, V, E)$ is deterministic. For any $a \in A$, define a partial map $\varphi_a : V \to V$ by $\varphi_a(u) = v$ if there exsists $(u, a, v) \in E$. We obtain the submonoid $M(G)$ of $\mathcal{PT}(V)$ generated by the set $\{\varphi_a\}_{a \in A}$, where $\mathcal{PT}(V)$ is the monoid of all partial maps $V \to V$. $M(G)$ is called the monoid of G.

Fix a deterministic A-graph $G = (A, V, E)$. Let i be an element of V, called an *initial* vertex of G. Let T be a subset of V, whose elements are called *terminal* vertices of G. We obtain a (unnecessarily finite) deterministic *automaton* $\mathcal{A}(G)$ in which V is a set of states, E is a set of edges, i is an initial state, and T is a set of terminal states.

Given edges $e_i = (u_i, a_i, u_{i+1})$ $(1 \leq i \leq n)$, the sequence $e_1 e_2 \cdots e_n$ is called *path* from a state u_1 to a state u_{n+1}. the word $a_1 a_2 \cdots a_n$ is a label of the path $p = e_1 e_2 \cdots e_n$, the length of p is n, and then we write it as $|p| = n$.

If u_1 is an initial state and v_n is a terminal state, then $e_1 e_2 \cdots e_n$ is called a *successful* path.

A deterministic automaton $\mathcal{A}(G)$ is called *accessible* if for any vertex v of G, there exists a path from a initial vertex to v.

A deterministic automaton $\mathcal{A}(G)$ is called *co-accessible* if for any vertex v of G, there exists a path from v to a terminal vertex.

Lemma 1. *For any deterministic automaton $\mathcal{A}$, there exists an accessible and co-accessible automaton $\mathcal{B}$ such that $L(\mathcal{A}) = L(\mathcal{B})$.*

There is an action of A^* on V, that is, we write as $vw = u$ if there exists a path from u to v with a label w.

Fix an automaton $\mathcal{A} = (A, V, E, I, T)$. Define a relation $\equiv$ on V defined by $v \equiv u$ $(u, v \in V)$ if and only if

$$\{w \in A^* | vw \in T\} = \{w \in A^* | uw \in T\}.$$

We get a new automaton $\overline{\mathcal{A}} = (A, \overline{V}, \overline{E}, \overline{I}, \overline{T})$, where $\overline{V} = V/\equiv$, $\overline{E} = \{(\overline{u}, a, \overline{v}) \mid (u, a, v) \in E\}$ (for $u \in V$), $\overline{u} = \{v \in V \mid u \equiv v\})$, $\overline{I} = I/\equiv$, $\overline{T} = T/\equiv$.

Lemma 2. *Let $\mathcal{A} = (A, V, E, I, T)$ be an deterministic accessible co-accessible automaton. Then $\overline{\mathcal{A}} = (A, \overline{V}, \overline{E}, \overline{I}, \overline{T})$ is a minimal automaton recognizing $L(\mathcal{A})$.*

Fix a deterministic A-graph $G = (A, V = \{v_1, v_2, \ldots\}, E)$. We get an minimal automaton $\mathcal{A}_G = (A', V', E', \{i\}, \{t\})$ where $A' = A \cup \{\alpha, \beta\}$, $V' = V \cup \{i, t\}$ and $E' = \{(i, \alpha, v_1), (v_j, \alpha, v_{j+1}), (v_{j+1}, \beta, v_j), (v_1, \beta, t) \mid j = 1, 2, \ldots\}$.

Theorem 1. *Let $G = (A, V, E)$ be a deterministic A-graph. For the automaton $\mathcal{A}_G$ constructed above, $M(G)$ is embedded in $Syn(L(\mathcal{A}_G))$.*

Consequently, any monoid is a submonoid of a syntactic monoid.

3 Embedding of inverse monoids in syntactic monoids

Definition 4. A monoid M is called an *inverse* monoid if for any $s \in M$, there exsists uniquely an element $m \in M$ with $msm = m, sms = s$.

Let $G = (A, V, E)$ be a deterministic A-graph. Then G is called *injective* if there is no pair of two edges of form (u, a, v) and (u', a, v), where $a \in A$, $u, u', v \in V$.

By choosing initial vertices and terminal vertices from V, we obtain an injective deterministic automaton $\mathcal{A}(G)$.

Then the monoid $M(G)$ of G is a submonoid of the symmetric inverse monoid $S(V)$ on the set of V.

Now we have the following results which are an inverse monoid-version of Lemma 2 and Theorem 1.

Lemma 3. *Let $\mathcal{A} = (A, V, E, I, T)$ be an deterministic accessible co-accessible injective automaton.*

Then $\overline{\mathcal{A}} = (A, \overline{V}, \overline{E}, \overline{I}, \overline{T})$ is a minimal automaton decognizing $L(\mathcal{A})$.

Fix a deterministic injective A-graph $G = (A, V = \{v_1, v_2, \ldots\}, E)$. We get an injective automaton $\mathcal{A}_G = (A', V', E', \{i\}, \{t\})$ where $A' = A \cup \{\alpha, \beta\}$, $V' = V \cup \{i, t\}$, $E' = \{(i, \alpha, v_1), (v_j, \alpha, v_{j+1}), (v_1, \alpha', i), (v_{j+1}, \alpha', v_j), (v_{j+1}, \beta, v_j), (v_1, \beta, t), (v_j, \beta', v_{j+1}), (t, \beta', v_1)$ $| j = 1, 2, \ldots\}$.

Theorem 2. *Let $G = (A, V, E)$ be a deterministic injectiveA-graph. For the automaton $\mathcal{A}_G$ constructed above, $M(G)$ is embedded in an inverse monoid $Syn(L(\mathcal{A}_G))$.*

Consequently, any inverse monoid is a submonoid of an inverse syntactic monoid.

4 Word problems for Syntactic monoids of context-free languages

Definition 5. Context-free languages are defined as languages consisting of words accepted by pushdown automata. Equivalently, context-free languages are defined languages accepted by formal grammars as follows :

A formal grammar Γ consists of a finite set V of symbols and a special symbol σ, a finite set of alphabets A and a subset P of $V^+ \times (V \cup A)^*$, which is called *product*. Then the formal grammar Γ is denoted by (V, A, P, σ).

Definition 6. Let L be a language over a finite alphabet A. Then a word problem for the syntactic monoid $Syn(L)$ is the following question:

For any pair of two words $w, w' \in A^*$, does there exists an algorithm deciding whether $(w, w') \in \sigma_L$ or $(w, w') \notin \sigma_L$?

Let I be a non-empty set of a semigroup S. Then I is called an *ideal* of S. An ideal I of S is called *completely prime* if for any $x, y \in S$, $xy \in I$ implies that either $x \in I$ or $y \in I$.

The following follows immediately.

Lemma 4. *Let L be a language over A and $sub(L)$ the set of subwords of words in L.*

Then the complement of $sub(L)$ in L is completely prime.

Corollary 1. *Let L be a language over A and $sub(L)$ the set of subwords of words in L.*

Then the syntactic monoid $Syn(L)$ has a zero element if and only if either $A^* \neq sub(L)$ or $A^* \neq sub(L^c)$.

Theorem 3. *Let L be a language over A. The syntactic monoid $Syn(L)$ has a zero element if and only if there exists a word w over A such that either $A^*wA^* \subseteq L$ or $A^*wA^* \subseteq L^c$.*

Problem 2 Let L be a deterministic context-free language over a finite alphabet A. Then is word problem for the syntactic monoid $Syn(L)$ undecidable ?

Problem 3 Let L be a deterministic context-free language over a finite alphabet A. Then is it decidable whether the syntactic monoid $Syn(L)$ has a zero element or not ?

5 Presentation of monoids with regular congruence classes

Result 4. *Let G be a finitely generated group and $\varphi : A^* \to G$ an onto homomorphism with $L = \varphi^{-1}(1)$ ($\subseteq A^*$). Then*

(1) ([6]) *G is finite if and only if L is a regular language.*

(2) ([7], [8], [9]) *G is vertually free (a finite extension of free group) if and only if L is a deterministic context-free language.*

Lemma 5. *Let L be a language of A^*. Then L is a union of σ_L-classes in A^*.*

Theorem 4. *Let L be a language of A^*. Then the following are equivalent :*

(1) *L is a σ_L-class in A^*.*

(2) $xLy \cap L \neq \emptyset \; (x, y \in A^*) \Rightarrow xLy \subseteq L.$

(3) *L is an inverse image $\phi^{-1}(m)$ of a homomorphism ϕ of A^* to a monoid M.*

Theorem 5. *For every finitely generated monoid M, there exist languages $\{L_m\}_{m\in M}$ of A^* such that M is embedded in the direct product of syntactic semigroups.*

Definition 7. Let M be a monoid and A a finite alphabet. M has the presentation with *regular congruence classes* if there exists a onto homomorphism of $\varphi : A^* \longrightarrow M$ is such that if for any $m \in M$, $\varphi^{-1}(m)$ is a regular language.

Definition 8. A monoid M is *residually finite* if for each pair of elements $m, m' \in M$, there exists a conguence μ on M such that the factor monoid M/μ is finite and $(m, m') \notin \mu$.

Theorem 6. *Let M be a finitely generated monoid and $\phi : A^* \longrightarrow M$ a onto homomorphism.*

Then for each $m \in M$, the following are equivalent.

(1) $\phi^{-1}(m)$ *is a regular language*

(2) $|M/\sigma_m| < \infty$.

Let M *be a monoid and* m *an element of* M. *Define a relation* σ_m *by* $a\sigma_m b$ $(a, b \in M)$ *if and only if*

$$\{(x,y) \in M \times M \mid xay = m\} = \{(x,y) \in M \times M \mid xby = m\}.$$

Then σ_m *is a congruence on* M.

Theorem 7. *Let* M *be a finitely generated monoid and* $\varphi : A^* \longrightarrow M$ *be a presentation of* M *with regular congruence classes. Then* M *is residually finite.*

References

[1] A. H. Clifford and G. B. Preston, *Algebraic theory of semigroups*, Amer. Math. Soc., Math. Survey, No.7, Providence, R.I., Vol.I(1961); Vol.II(1967).

[2] S. Eilenberg, *Automata, Languages, and Machines*, Vol. B Academic Press, 1974.

[3] J. E. Hopcroft and J. D. Ullman *Introduction to Automata theory, Languages, and Computation*, Addison-Wesly Publishing, 1979.

[4] G. Lallemant, *Semigroups and combinatorial applications*, Jhon Wiley & Sons, 1979.

[5] J. E. Pin, *Varieties of Formal Languages*, North Oxford Academic Publishers, 1984.

[6] M. J. Dunwoody, *The accessiblity of finitely presentesd groups*, Invnt. Math., **81**(1985), 449-457.

[7] V. A. Anisimov, *Groups languages*, Kibernetika, **4**(1971), 18-24.

[8] D. C. Muller and P. E. Shupp, *Groups, the theory of ends, and context-free languages*, J. Comput. System Sci., **67**(1983), 295-310.

[9] D. C. Muller and P. E. Shupp, *The theory of ends, pushdown automata, and second-order logic*, Thoeritical Comput. Sci., **37**(1985), 51-75.