数理解析研究所講究録
1166 巻 2000 年 95-99

A note on the languages recognized by commutative asynchronous automata*

B. Imreh[†] M. Ito[‡] A. Pukler[§]

Abstract

The languages recognized by commutative asynchronous automata are studied and described here. It turns out that over a finite nonvoid alphabet X with $|X| = k$, the languages recognized by commutative asynchronous automata constitute such a Boolean algebra which is isomorphic to the Boolean algebra consisting of all subsets of the set $\{0,1\}^k$.

1 Introduction

The decomposition of commutative asynchronous automata is studied in [1] and it is proved that every commutative asynchronous automaton can be embedded isomorphically into a suitable quasi-direct power of a two-state commutative asynchronous automaton. Moreover, the directable commutative asynchronous automata are also investigated in [1], and it is shown that the exact bound for the maximal length of minimum-length directing words

*This work has been supported by the Japanese Ministry of Education, Mombusho International Scientific Research Program, Joint Research 10044098, the Hungarian National Foundation for Science Research, Grant T030143, and the Ministry of Culture and Education of Hungary, Grant FKFP 0704/1997.

†Department of Informatics, University of Szeged, Árpád tér 2, H-6720 Szeged, Hungary

‡Department of Mathematics, Faculty of Science, Kyoto Sangyo University, Kyoto 603-8555, Japan

§Department of Computer Science, István Széchenyi College, Hédervári út 3., H-9026 Győr, Hungary

of commutative asynchronous automata of n states is equal to $n-1$, *i.e.*, the exact bound is the same as in the commutative case (see *eg.* [3] or [4]). Surprisingly, the exact bound decreases drastically to $[\log_2(n)]$ if we consider only such elements of this class which are generated by one element. Paper [2] deals with the decomposition of commutative asynchronous nondeterministic automata. Here, we study now the languages recognized by commutative asynchronous automata. It turns out that there are a few of them, and they constitute a Boolean algebra under a fixed alphabet.

2 Preliminaries

We recall here a few notions and notation necessary in the sequal. Let X be a nonempty alphabet with $|X| = k$. Without loss of generality, we may assume that $X = \{x_1, \ldots, x_k\}$. Throughout this paper we shall work uder this fixed alphabet X. The set of all finite words over X is denoted by X^*. For the length of a word $p \in X^*$, we use the notation $|p|$. For any $p \in X^*$, let us denote by alph(p) the set of the all letters occuring in the word p. One can extend the function alph to languages in a natural way. The *shuffle product* of two words $u, v \in X^*$ is the set

$$u \diamond v = \{w : w = u_1 v_1 \ldots u_n v_n,\ u = u_1 \ldots u_n,\ v = v_1 \ldots v_n,\ u_i.v_j \in X^*\}.$$

The shuffle product can be extended to languages as well. We use the *Parikh mapping* denoted by Ψ. For its definitions, let $N = \{0, 1, 2, \ldots\}$, and let us define the mapping $\Psi : X^* \to N^k$, by

$$\Psi(u) = (\mu_{x_1}(u), \ldots, \mu_{x_k}(u)),$$

where $\mu_{x_j}(u)$ denotes the number of the occurrences of x_j in u, for every j, $j = 1, \ldots, k$.

By *automaton* or X-*automaton* we mean a system $\mathbf{A} = (A, X)$, where A is the finite nonvoid set of *states*, X is the finite nonempty set of *input signs*, and every input sign $x \in X$ is realized as a unary operation $x^{\mathbf{A}} : A \to A$. The automaton $\mathbf{A} = (A, X)$ is *commutative* if $a(xy)^{\mathbf{A}} = a(yx)^{\mathbf{A}}$ is valid, for all $a \in A$ and $x, y \in X$. Another particular automata are the asynchronous ones. $\mathbf{A}$ is called *asynchronous* if $ax^{\mathbf{A}} = a(xx)^{\mathbf{A}}$, for all $a \in A$ and $x \in X$.

Some particular commutative asynchronous automata introduced in [1] will be used in the following section.

For every $n \geq 1$, let us define the automaton $\mathbf{H}_n = (\{0,1\}^n, \{x_1, \ldots, x_n\})$ in the following way. For all $(i_1, \ldots, i_n) \in \{0,1\}^n$ and $x_j \in \{x_1, \ldots, x_n\}$, let

$$(i_1, \ldots, i_n)x_j^{\mathbf{H}_n} = \begin{cases} (i'_1, \ldots, i'_n) & \text{if } i_j = 0, \text{ where } i'_t = i_t,\ t = 1, \ldots, n,\ t \neq j, \\ & \text{and } i'_j = 1, \\ (i_1, \ldots, i_n) & \text{otherwise.} \end{cases}$$

The automaton $\mathbf{H}_n$ can be visualized as follows. Its states are the vertices of the n-dimensional hyper-cube and any input sign takes the automaton from a vertex into its neighbour or fixes the state given. Moreover, x_j changes only the jth component. By the definition of $\mathbf{H}_n$, it is easy to see that $\mathbf{H}_n$ is commutative and asynchronous.

A *recognizer* or X-*recognizer* is a system $\mathcal{A} = (\mathbf{A}, a_0, F)$ which consists of an X-automaton $\mathbf{A}$, an *initial state* $a_0 \in A$, and a set $F(\subseteq A)$ of *final states.* The language *recognized* by $\mathcal{A}$ is

$$L(\mathcal{A}) = \{w : w \in X^* \text{ and } a_0 w^{\mathbf{A}} \in F\}.$$

It is also said that $L(\mathcal{A})$ is *recognizable* by the automaton $\mathbf{A}$.

3 Results

For every k dimensional binary vector $\mathbf{i} = (i_1, \ldots, i_k)$, a language $L_{\mathbf{i}}$ over X can be defined as follows. Let

$$L_{\mathbf{i}} = \Psi^{-1}(\mathbf{i}) \diamond (\mathrm{alph}(\Psi^{-1}(\mathbf{i}))^*.$$

Moreover, if $B \subseteq \{0,1\}^k$, then we can define the language L_B by

$$L_B = \cup_{\mathbf{i} \in B} L_{\mathbf{i}}.$$

The languages L_B, $B \subseteq \{0,1\}^k$ are strongly related to the languages recognizable by commutative asynchronous X-automata. This strong relationship is presented by the following statement.

Proposition 1. *A language $L \subseteq X^*$ is recognized by a commutative asynchronous X-automaton if and only if $L = L_B$ for some $B \subseteq \{0,1\}^k$.*

Proof. Let $L \subseteq X^*$ be an arbitrary language and let us suppose that L can be recognized by a recognizer $\mathcal{A} = (\mathbf{A}, a_0, F)$, where $\mathbf{A} = (A, X)$ is a commutative asynchronous X-automaton. Let us observe that $ap^{\mathbf{A}} = a(x_{i_1} \dots x_{i_s})^{\mathbf{A}}$, $1 \leq s \leq k$ is valid for every $p \in X^*$ with $\text{alph}(p) = \{x_{i_1}, \dots, x_{i_s}\}$ since $\mathbf{A} = (A, X)$ is commutative and asynchronous. By the commutativity, we may suppose that $i_1 < i_2 < \dots < i_s$. Therefore, for every $p \in L$, there exists a uniquely determined word $x_{i_1} \dots x_{i_s}$ such that $a_0 p^{\mathbf{A}} = a_0(x_{i_1} \dots x_{i_s})^{\mathbf{A}}$. Now, let us denote by K the subset of L which consists of all words q in L for which $|q| = |\text{alph}(q)|$ and if $q = x_{i_1} \dots x_{i_s}$, then $i_1 < i_2 < \dots < i_s$. Then it is easy to see that

$$L = \bigcup_{q \in K} (\Psi^{-1}(\Psi(q))) \diamond (\text{alph}(q))^*.$$

On the other hand, by the definition of K, the mapping μ which is defined by $\mu : q \to \Psi(q)$, $q \in K$, is a one-to-one mapping of the language K into $\{0,1\}^k$. Consequently, if the image of K under μ is denoted by B, then $B \subseteq \{0,1\}^k$, moreover,

$$L = \bigcup_{q \in K} (\Psi^{-1}(\Psi(q))) \diamond (\text{alph}(q))^* = \bigcup_{\mathrm{i} \in B} \Psi^{-1}(\mathrm{i}) \diamond (\text{alph}(\Psi^{-1}(\mathrm{i}))^* = \cup_{\mathrm{i} \in B} L_{\mathrm{i}} = L_B.$$

and consequently, $L = L_B$. In particular, if $L = \emptyset$, then $B = \emptyset$.

Conversely, let $L = L_B = \cup_{\mathrm{i} \in B} L_{\mathrm{i}}$ for some $B \subseteq \{0,1\}^k$. Then it is easy to prove that the commutative asynchronous automaton $\mathbf{H}_k$ based on the k dimensional hyper-cube recognizes L by $(\mathbf{H}_k, (0, 0, \dots, 0), B)$, and thus, L can be recognized by a commutative asynchronous X-automaton.

From the description of the languages over X, recognized by commutative asynchronous X-automata, it follows that these languages are closed under the union and intersection. What is more that is presented by the following assertion.

Proposition 2. *The number of the languages over* $X = \{x_1, \ldots, x_k\}$*, which can be recognized by commutative asynchronous* X*-automata, is equal to* 2^{2^k}*, moreover, these languages constitute a Boolean algebra which is isomorphic to the Boolean algebra consisting of all the subsets of the set* $\{0,1\}^k$*.*

Proof. Let us denote by $\mathcal{L}_X$ the set of languages, recognized by commutative asynchronous X-automata. Let $L \in \mathcal{L}_X$ be an arbitrary language. By the proof of Proposition 1, there exists a $B \subseteq \{0,1\}^k$ such that $L = L_B$. Therefore, to every language $L \in \mathcal{L}_X$, we can assign a subset B of $\{0,1\}^k$. Let us denote this mapping by φ. Then φ is a mapping of $\mathcal{L}_X$ into $\{0,1\}^k$. On the other hand, in the proof of Proposition 1 it is shown that for every $B \subseteq \{0,1\}^k$, there exists a language $L \in \mathcal{L}_X$ such that $L \doteq L_B$, and therefore, φ is surjective. Finally, it is easy to see that if $L_1 \neq L_2 \in \mathcal{L}_X$, then $L_1\varphi \neq L_2\varphi$.

Consequently, φ is a one-to-one mapping of $\mathcal{L}_X$ onto $\{0,1\}^k$. Moreover, it is evident that $(L_1 \cup L_2)\varphi = L_1\varphi \cup L_2\varphi$, $(L_1 \cap L_2)\varphi = L_1\varphi \cap L_2\varphi$, and $\bar{L}_1\varphi = \overline{L_1\varphi}$, for all $L_1, L_2 \in \mathcal{L}_X$, where $\bar{L}$ and $\bar{L}\varphi$, denotes the corresponding complements, respectively. Consequently, φ is an isomorphism. This isomorphism provides that $|\mathcal{L}_X| = 2^{2^k}$. This ends the proof of Proposition 2.

References

[1] Imreh, B., M. Ito, A. Pukler, On commutative asynchronous automata, Proceedings of The Third International Colloquium on Words, Languages, and Combinatorics, Kyoto, 2000, to appear.

[2] Imreh, B., M. Ito, A. Pukler, On commutative asynchronous nondeterministic automata, *Acta Cybernetica*, to appear.

[3] Imreh, B., M. Steinby, Some remarks on directable automata, *Acta Cybernetica* **12** (1995), 23-35.

[4] Rystsov, I., Exact linear bound for the length of reset words in commutative automata, *Publicationes Mathematicae* **48** (1996), 405-409.