

# RIGHT CONGRUENCES FOR $\omega$ -REGULAR LANGUAGES

Hideki YAMASAKI

(山崎 秀記, 一橋大学)

*Department of Mathematics, Hitotsubashi University  
Kunitachi, Tokyo 186 Japan*

March 9, 1995

## 1 Introduction

Let  $\Sigma$  be an alphabet and  $\Sigma^*$  be a free monoid generated by  $\Sigma$ . One of the main feature of the study of regular languages (of finite words) over  $\Sigma$  is the study of the right congruences (i.e., equivalence relations preserved under the concatenation from right) of  $\Sigma^*$ .

The next theorem is well known.

**Theorem 1 (Myhill-Nerode)** *The following three conditions for a language  $L \subseteq \Sigma^*$  is equivalent.*

- (1)  *$L$  is regular.*
- (2)  *$L$  is a union of some equivalence classes of a right congruence of finite index.*
- (3) *The equivalence relation  $\sim$  defined by :*

$$u \sim v \text{ if for any } x \in \Sigma^* \text{ } ux \in L \Leftrightarrow vx \in L$$

*is a right congruence of finite index.*

Moreover, for any regular language  $L$ , there exists a one-to-one correspondence between finite automata accepting  $L$  and right congruences of finite index recognizing  $L$  in the sense of (2) of the Myhill-Nerode's theorem.

In the case of  $\omega$ -regular languages, the situation is not so simple as the case of regular languages. As shown in Example 1 below, there exists an  $\omega$ -regular language  $L$  which does not have a unique deterministic minimum automaton accepting  $L$ . For the syntactic characterization of  $\omega$ -regular languages, the results using (two-sided) congruence was obtained by Arnold [1], and the syntactic right congruence can recognize only the  $\omega$ -languages in the subclass of  $\omega$ -regular languages [4, 5].

Recently, Do Long Van, B.Le Saïc and I.Litovsky [3] give necessary and sufficient conditions for finite right congruences to recognize  $\omega$ -regular languages. Maler and Staiger [4] introduce a notion of a family of right congruences, called a FORC, and show that an  $\omega$ -language  $L$  is regular if and only if it is saturated by a finite FORC.

In this paper, we define simple and normal FORCs, and show that for any  $\omega$ -language  $L$ ,  $L$  is accepted by a deterministic Büchi (Muller, respectively) automaton if and only if it is covered (saturated) by a simple (normal) FORC. Moreover, there exists a one-to-one correspondence between simple (normal) FORCs covering (saturating)  $L$  and deterministic Büchi (Muller) automata accepting  $L$ .

## 2 Basic Definitions

For an alphabet  $\Sigma$ , we call a mapping  $\alpha \in \Sigma^{\mathbb{N}}$  an  $\omega$ -word over  $\Sigma$ , and write  $\alpha = a_0 a_1 a_2 \dots$  where  $a_n = \alpha(n)$  for each  $n$ . The set of all  $\omega$ -words over  $\Sigma$  is denoted by  $\Sigma^\omega$ , and that of all finite words over  $\Sigma$  is denoted by  $\Sigma^*$ , as usual. For  $u = a_0 a_1 a_2 \dots a_m \in \Sigma^*$ , we also denote the  $(n+1)$ th letter  $a_n$  of  $u$  as  $u(n)$ ,  $0 \leq n \leq m$ .

The concatenation operation and prefix relation on  $\Sigma^*$  are generalized as follows. For  $u \in \Sigma^*$  and  $\alpha \in \Sigma^\omega$ ,  $u\alpha$  is defined to be the  $\omega$ -word obtained by concatenating  $u$  before  $\alpha$ . If  $\beta = u\alpha$ , then we say that  $u$  is a *prefix* of  $\beta$ . For any  $u \in \Sigma^*$  and  $\alpha \in \Sigma^* \cup \Sigma^\omega$ , we write  $u \preceq \alpha$  if  $u$  is a prefix of  $\alpha$ .

For  $K \subseteq \Sigma^*$  and  $L \subseteq \Sigma^\omega$ , we define  $KL = \{u\alpha \mid u \in K \text{ and } \alpha \in L\}$  and  $K^\omega = \{v_1 v_2 \dots \mid v_1, v_2, \dots \in K - \{\epsilon\}\}$ , where  $v_1 v_2 \dots$  is the  $\omega$ -word obtained by concatenating  $v_1, v_2, \dots$  one after another.

For  $\alpha \in \Sigma^* \cup \Sigma^\omega$ , we define  $\underline{\alpha} = \{a \mid a = \alpha(n) \text{ for some } n\}$ , and  $\underline{\underline{\alpha}} = \{a \mid a = \alpha(n) \text{ for infinitely many } n\}$ . That is,  $\underline{\alpha}$  is the set of letters appearing in  $\alpha$  and  $\underline{\underline{\alpha}}$  is the set of letters appearing infinitely many times in  $\alpha$ .

A deterministic finite automaton over  $\Sigma$  is a quadruple  $A = (Q, \Sigma, \delta, s)$ , with the finite set  $Q$  of states, the input alphabet  $\Sigma$ , the transition function  $\delta : Q \times \Sigma \rightarrow Q$ , and the initial state  $s \in Q$ . (We do not include the usual set of accepting states in this definition.) For any  $\alpha \in \Sigma^* \cup \Sigma^\omega$ , the run  $Run(A, \alpha)$  of  $A$  over  $\alpha$  is the  $\rho \in Q^* \cup Q^\omega$  such that  $\rho(0) = s$  and  $\rho(i+1) = \delta(\rho(i), \alpha(i))$  for any  $i$ .

For a deterministic finite automaton  $A = (Q, \Sigma, \delta, s)$  and the set  $F \subseteq Q$  of accepting states, the  $\omega$ -language  $I(A, F)$  accepted by  $(A, F)$  is defined by:

$$I(A, F) = \{\alpha \mid \underline{Run(A, \alpha)} \cap F \neq \phi\}.$$

The automaton  $(A, F)$  is called a Büchi automaton.

For a deterministic finite automaton  $A = (Q, \Sigma, \delta, s)$  and the set  $\mathbf{F} \subseteq 2^Q$  of accepting sets of states, the  $\omega$ -language  $R(A, \mathbf{F})$  accepted by  $(A, \mathbf{F})$  is defined by:

$$R(A, \mathbf{F}) = \{\alpha \mid \underline{Run(A, \alpha)} \in \mathbf{F}\}.$$

The automaton  $(A, \mathbf{F})$  is called a Muller automaton.

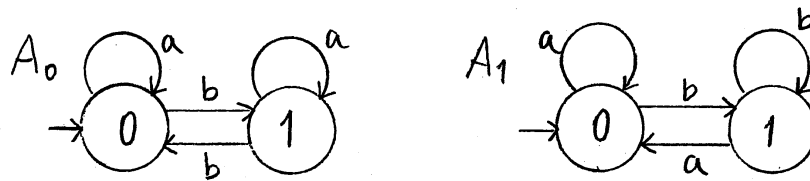
We define

$$\mathbf{I} = \{I(A, F) \mid (A, F) \text{ is a Büchi automaton over } \Sigma\},$$

$$\mathbf{R} = \{R(A, \mathbf{F}) \mid (A, \mathbf{F}) \text{ is a Muller automaton over } \Sigma\}.$$

That is,  $\mathbf{I}$  ( $\mathbf{R}$ , respectively) is the class of  $\omega$ -languages accepted by Büchi (Muller) automata over  $\Sigma$ . The class  $\mathbf{R}$  is called the class of  $\omega$ -regular languages over  $\Sigma$ . It is shown [2, 5, 6] that  $\mathbf{I} \subset \mathbf{R}$ , and  $L \in \mathbf{R}$  if and only if  $L = \bigcup_{i=1, n} J_i K_i^\omega$  for some regular languages  $J_i, K_i \subseteq \Sigma^*$  ( $i = 1, \dots, n$ ).

**Example 1** Let  $\Sigma = \{a, b\}$ . The  $\omega$ -language  $(\Sigma^*b)^\omega$  is accepted by two essentially different two state Muller automata  $(A_i = (\{0, 1\}, \Sigma, \delta_i, 0), \mathbf{F}_i)$  ( $i = 0, 1$ ) with  $\delta_0(p, a) = p, \delta_0(p, b) = 1-p$  for any  $p = 0, 1, \mathbf{F}_0 = \{\{0, 1\}\}, \delta_1(p, a) = 0, \delta_1(p, b) = 1$  for any  $p = 0, 1$  and  $\mathbf{F}_1 = \{\{1\}, \{0, 1\}\}$ .



Note that  $(\Sigma^*b)^\omega$  is also accepted by the Büchi automaton  $(A_1, \{1\})$ .

A *right congruence*  $\sim$  of  $\Sigma^*$  is an equivalence relation preserved under the concatenation from right, that is,  $u \sim v$  implies  $ux \sim vx$  for any  $x \in \Sigma^*$ . A right congruence is said to be *finite* if it has a finite number of equivalence classes.

Let  $\sim$  be a right congruence of  $\Sigma^*$  and  $u \in \Sigma^*$ . The equivalence class of  $\sim$  containing  $u$  is denoted by  $[u]_{\sim}$ , and we simply write  $[u]$  if the right congruence  $\sim$  is clear from the context.

For any finite right congruence  $\sim$ , we can assign a deterministic finite automaton  $A_{\sim} = (\Sigma^*/\sim, \Sigma, \delta_{\sim}, [\epsilon])$ , with  $\delta_{\sim}([u], a) = [ua]$  for every  $u \in \Sigma^*$  and  $a \in \Sigma$ . Conversely, for any deterministic finite automaton  $A = (Q, \Sigma, \delta, s)$ , we can assign a finite right congruence  $\sim^A$  defined by:  $u \sim^A v$  if and only if  $\delta(s, u) = \delta(s, v)$ . Note that these establish the one to one correspondence between finite automata and right congruences, i.e.,  $\sim^A \sim = \sim$  and  $A_{\sim^A}$  is isomorphic to  $A$ .

### 3 Simple FORCs and Normal FORCs

Recently, Maler and Staiger [4] defined a family of finite right congruences, called a FORC, to study the  $\omega$ -languages. A FORC  $C$  is a family of finite right congruences  $C = (\sim, \{\sim_{[u]} \mid u \in \Sigma^*\})$  such that for any  $u, x, y \in \Sigma^*$ ,  $x \sim_{[u]} y$  implies  $ux \sim uy$ . The right congruence  $\sim$  is called the leading right congruence of  $C$ . We write  $\sim_u$  for  $\sim_{[u]}$  and  $[x]_u$  for  $[x]_{\sim_u}$ . Thus  $\sim_u = \sim_v$  and  $[x]_u = [x]_v$  for any  $x \in \Sigma^*$ , if  $u \sim v$ .

We say that a FORC is *simple* if  $x \sim_u y$  if and only if  $ux \sim uy$  for any  $u, x, y \in \Sigma^*$ . In this case the FORC is determined by the leading right congruence  $\sim$ , so we call the FORC a simple FORC induced by  $\sim$ .

We say that a FORC is *normal* if  $x \sim_u y$  if and only if

- (1)  $ux \sim uy$  and
- (2)  $\{[uv] \mid v \preceq x \text{ and } uv \sim uxz \text{ for some } z\}$   
 $= \{[uv] \mid v \preceq y \text{ and } uv \sim uyz \text{ for some } z\}$ .

In this case the FORC is determined by the leading right congruence  $\sim$ , so we call the FORC a normal FORC induced by  $\sim$ .

An  $\omega$ -language  $L$  is *covered* by a FORC  $C = (\sim, \{\sim_{[u]} \mid u \in \Sigma^*\})$  if  $L$  is a finite union of  $\omega$ -languages of the form  $[u][v]_u^{\omega}$  with  $u \sim uv$ . An  $\omega$ -language  $L$  is *saturated* by a FORC  $C = (\sim, \{\sim_{[u]} \mid u \in \Sigma^*\})$  if  $[u][v]_u^{\omega} \cap L \neq \emptyset$  implies  $[u][v]_u^{\omega} \subseteq L$ .

The following lemma proved in [4] assures that any FORC covers all of  $\omega$ -words.

**Lemma 2 ([4])** For any FORC  $C$  over  $\Sigma$ ,  $\bigcup_{u \sim uv} [u][v]_u^\omega = \Sigma^\omega$ .

**Lemma 3** Any FORC  $C$  saturating  $L$  covers  $L$ .

**Proof.** Let  $K = \bigcup\{[u][v]_u^\omega \mid [u][v]_u^\omega \cap L \neq \emptyset\}$ .  $L \subseteq K$  is clear from the definition of  $K$  and Lemma 2. Since  $C$  saturates  $L$ ,  $K \subseteq L$ .  $\square$

The converse of the above lemma does not always hold, as shown in the Example 2 below.

**Example 2** Let  $\Sigma = \{a, b\}$  and  $\sim$  be a right congruence with the equivalent classes  $\{\epsilon \cup \Sigma^*a, \Sigma^*b\}$ . Then the simple FORC induced by  $\sim$  is  $(\sim, \{\sim_\epsilon, \sim_b\})$ , where  $\sim_\epsilon = \sim$  and  $\sim_b$  has the equivalent classes  $\{\epsilon \cup \Sigma^*b, \Sigma^*a\}$ . Thus,  $(\Sigma^*b)^\omega = [b][\epsilon]_b$  is covered by  $C$ . Since  $(\Sigma^*b)^\omega \cap [\epsilon][\epsilon]_\epsilon = (\Sigma^*b)^\omega \cap (\Sigma^*a)^\omega \neq \emptyset$  and  $(\Sigma^*a)^\omega - (\Sigma^*b)^\omega \neq \emptyset$ ,  $(\Sigma^*b)^\omega$  is not saturated by  $C$ .

If a FORC  $C$  is normal, then it saturates any  $\omega$ -languages covered by  $C$ .

**Lemma 4** If a FORC  $C$  is normal,  $C$  saturates  $L$  if and only if  $C$  covers  $L$ .

**Proof.** It is enough to show that a normal FORC  $C$  saturating  $L$  covers  $L$ . Let  $C = (\sim, \{\sim_u \mid u \in \Sigma^*\})$  be a normal FORC. For any  $u, v$  such that  $u \sim uv$ , we show that  $\alpha \in [u][v]_u^\omega$  if and only if  $\underline{\text{Run}}(A_\sim, \alpha) = \{[uz] \mid z \preceq v\}$ .

It is easy to see that if  $\alpha \in [u][v]_u^\omega$  then  $\underline{\text{Run}}(A_\sim, \alpha) = \{[uz] \mid z \preceq v\}$ . Assume  $\underline{\text{Run}}(A_\sim, \alpha) = \{[uz] \mid z \preceq v\}$ . Then there exists  $x, y_1, y_2, \dots \in \Sigma^*$  such that  $\alpha = xy_1y_2\dots$ ,  $u \sim x \sim xy_i$  for any  $i$ , and  $\{[uz] \mid z \preceq v\} = \{[xz] \mid z \preceq y_i\}$  for any  $i$ . It means that  $v \sim_u y_i$  for any  $i$ . Thus,  $\alpha \in [u][v]_u^\omega$ .

Now, assume  $\alpha \in [u_1][v_1]_{u_1}^\omega \cap [u_2][v_2]_{u_2}^\omega$ . It means that  $\underline{\text{Run}}(A_\sim, \alpha) = \{[u_1x] \mid x \preceq v_1\} = \{[u_2x] \mid x \preceq v_2\}$  and  $[u_1][v_1]_{u_1}^\omega = [u_2][v_2]_{u_2}^\omega$ . Hence  $C$  saturates any  $\omega$ -languages covered by  $C$ .  $\square$

**Lemma 5** If  $L$  is covered by a simple FORC induced by  $\sim$ , then  $L = \bigcup_{i=1}^n [u_i][\epsilon]_{u_i}$  for some  $u_1, \dots, u_n$ .

**Proof.** Let  $(\sim, \{\sim_u \mid u \in \Sigma^*\})$  be a simple FORC induced by  $\sim$ . If  $u \sim uv$ , then  $\epsilon \sim_u v$ . Thus,  $[u][v]_u = [u][\epsilon]_u$  for any  $u \sim uv$ .  $\square$

## 4 Main Results

Now we show the main results of this paper.

**Theorem 6** *An  $\omega$ -language  $L$  is in the class **I** if and only if it is covered by a simple FORC. Moreover, there exists a one-to-one correspondence between Büchi automata accepting  $L$  and simple FORCs covering  $L$ .*

**Proof.** Let  $(A = (Q, \Sigma, \delta, s), F)$  be a Büchi automaton such that  $L = I(A, F)$ . Then

$$L = \bigcup_{q \in F} \{u \mid \delta(s, u) = q\} \{v \mid \delta(q, v) = q\}^\omega.$$

Consider the simple FORC induced by  $\sim^A$ . It is clear that  $x \sim_u^A y$  if and only if  $\delta(s, ux) = \delta(s, uy)$ . Thus,

$$L = \bigcup_{\delta(s, u) \in F} [u][\epsilon]_u^\omega$$

Hence,  $L$  is covered by the simple FORC induced by  $\sim^A$ .

To show the converse, consider a simple FORC induced by  $\sim$ , and let  $L = \bigcup_{i=1}^n [u_i][\epsilon]_{u_i}^\omega$ . We define the Büchi automaton  $(A_\sim, F)$  with  $F = \{[u_i] \mid i = 1, \dots, n\}$ . Then an  $\omega$ -word  $\alpha$  is in  $L$  if and only if  $\alpha \in [u_i][\epsilon]_{u_i}^\omega$  for some  $i$  if and only if  $\alpha$  is accepted by  $(A_\sim, F)$ . Thus,  $L = I(A_\sim, F)$ .  $\square$

**Theorem 7** *An  $\omega$ -language  $L$  is in the class **R** if and only if it is saturated by a normal FORC. Moreover, there exists a one-to-one correspondence between Muller automata accepting  $L$  and normal FORCs saturating  $L$ .*

**Proof.** Let  $(A = (Q, \Sigma, \delta, s), F)$  be a Muller automaton and  $L = R(A, F)$ . We define  $run(q, a_1 \dots a_n)$  is a finite sequence  $q_0 \dots q_n$  of states such that  $q_0 = q$  and  $q_i = \delta(q_{i-1}, a_i)$  for all  $i, i = 1, \dots, n$ . Then

$$L = \bigcup_{q \in F \in \mathbf{F}} \{u \mid \delta(s, u) = q\} \{v \mid \delta(q, v) = q \text{ and } \underline{run}(q, v) = F\}^\omega$$

Consider the normal FORC induced by  $\sim^A$ . Then, for any  $u, x, y$  such that  $ux \sim^A uy \sim^A u$ ,  $x \sim_u^A y$  if  $\underline{run}(\delta(s, u), x) = \underline{run}(\delta(s, u), y)$ . It is easy to see that

$$L = \bigcup \{[u][v]_u^\omega \mid \delta(s, u) = \delta(s, uv) \text{ and } \underline{run}(\delta(s, u), v) \in \mathbf{F}\}$$

Hence,  $L$  is covered by the normal FORC induced by  $\sim^A$ . Since the FORC is normal, it saturates  $L$ .

To show the converse, consider the normal FORC induced by  $\sim$ , and let  $L = \bigcup_{i=1}^n [u_i][v_i]_{u_i}^\omega$  with  $u_i v_i \sim u_i$ . We construct the Muller automaton  $(A_\sim, \mathbf{F})$ , where  $\mathbf{F} = \{F_i \mid i = 1, \dots, n\}$ , and  $F_i = \{[u_i z] \mid z \preceq v_i\}$  for any  $i$ . It is clear that  $L \subseteq R(A, \mathbf{F})$ .

Assume  $\alpha \in R(A, \mathbf{F})$ . Then  $\underline{\text{Run}}(A, \alpha) = F_i$  for some  $i$ . Since  $[u_i] \in F_i$ ,  $\alpha$  can be written as  $\alpha = x y_1 y_2 \dots$  so that  $x \sim u_i$  and  $y_j \sim_{u_i} v_i$  for all  $j$ . Thus,  $\alpha \in L$ .  $\square$

## References

- [1] A. Arnold, A syntactic congruence for rational  $\omega$ -languages, *T.C.S.* 39 (1985) 333–335.
- [2] J.R.Büchi, On a decision method in restricted second-order arithmetic, *Logic, Methodology and Philosophy of Science* (Stanford Univ. Press, 1960) 1–11.
- [3] Do Long Van, B.Le Saëc and I.Litovsky, A syntactic approach to deterministic  $\omega$ -automata, in *Journées Franco-Berges: Automata theory and applications*, Rouen (1991).
- [4] O.Maler and L.Staiger, On syntactic congruence for  $\omega$ -regular languages, *L.N.C.S.* 665 (1993) 586 – 594.
- [5] L.Staiger, Finite State  $\omega$ -languages *J.C.S.S.* 27 (1983) 434–448.
- [6] M.Takahashi and H.Yamasaki, A note on  $\omega$ -regular languages, *T.C.S.* 23 (1983) 217–225.