Equivalence Problems for Mappings on Infinite Strings

by

Karel Culik II and Jan Pachl

Research Report CS-80-41
Department of Computer Science
University of Waterloo

September 1980

Faculty of Mathematics

University of Waterloo
Waterloo, Ontario, Canada
N2L 3G1

Equivalence problems for mappings on infinite strings*

by

Karel Culik II and Jan Pachl

Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada
N2L 3G1

* This research was supported by the National Sciences and Engineering Council of Canada, under Grant No. A 7403.

Abstract

This paper is concerned with sets of infinite strings (ω-languages) and mappings between them. The main result is that there is an algorithm for testing the (string by string) equality of two homomorphisms on an ω-regular set of infinite strings. As a corollary we show that it is decidable whether two functional finite-state transducers define the same function on infinite strings (are ω-equivalent).

0. Introduction

Infinite strings and sets of them (ω-languages) have been extensively studied, see e.g. Eilenberg (1974), Cohen and Gold (1977). Finite transducers on infinite strings were considered by Boasson and Nivat (1979).

Our main interest here is testing the equivalence of finite transducers on infinite strings. We consider finite transducers with accepting states (for precise definition see Section 2). It is known that for finite transducers (rational transductions) the equivalence on finite strings is undecidable in general but decidable for functional transducers (Berstel, 1979). The latter result holds even when the domain is restricted to a context free language (Culik 1979). Our main goal is to prove analogous results for finite transducers working on infinite strings (but we do not consider any domain restriction). The equivalence of two finite transducers on finite strings implies also their ω-equivalence, however the converse does not hold since two ω-equivalent transducers might produce their outputs at different speeds. Thus ω-equivalence is a necessary condition for equivalence but does not easily reduce to it.

In Section 2 we show some auxiliary results on deterministic ω-regular languages. In the next section we extend the techniques from Salomaa (1978), Culik and Salomaa (1978) and Culik (1979) and show that the (string by string) equivalence of two homomorphisms on an ω-regular set is decidable. Then we show that the ω-equivalence problem for functional transducers reduces to it. Finally, we note that the undecidability of the ω-equivalence for nondeterministic finite transducers (or gsm) easily follows from the undecidability of their equivalence.

1. Preliminaries

Our basic terminology is a mixture of (Eilenberg, 1974) and (Cohen and Gold, 1977).

For a finite alphabet Σ, let Σ^* be the set of finite strings over Σ, Σ^ω the set of infinite strings over Σ, and $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$. The empty string is denoted by ε, the length of a string w in Σ^* by $|w|$.

We consider two classes of subsets of Σ^ω (ω-languages): ω-regular and deterministic ω-regular ω-languages. We write a (nondeterministic) <u>finite state automaton</u> as $M = (K, \Sigma, \delta, q_0, F)$, where K is a finite set of states, Σ is the input alphabet, $\delta : K \times \Sigma \to 2^K$ is the transition function, $q_0 \in K$ is the initial state and $F \subseteq K$ is the set of final states. For $\alpha \in \Sigma^\omega$, $\alpha = a_0a_1a_2\cdots$, and an infinite sequence $r = p_0p_1p_2\cdots$ of states $p_i \in K$, we say that r is a <u>run of M on α</u> if $p_0 = q_0$ and $p_i \in \delta(p_{i-1}, a_{i-1})$ for $i \geq 1$. For an infinite sequence of states $r = p_0p_1p_2\cdots$, the set of states that appear infinitely many times in r is denoted by $\mathrm{INS}(r)$. The <u>ω-language accepted by automaton M</u> is defined as

$$L^\omega(M) = \{\alpha \in \Sigma^\omega \mid \text{there is a run } r \text{ on } \alpha \text{ such that } \mathrm{INS}(r) \cap F \neq \phi\}.$$

In the notation of Cohen and Gold (1977), $L^\omega(M) = T_2(M)$; in the notation of Eilenberg (1974), $L^\omega(M) = \|M\|$.

An ω-language is <u>ω-regular</u> (<u>deterministic ω-regular</u>) if $L = L^{\omega}(M)$ for some finite state (deterministic finite state) automaton M. We refer to Eilenberg (1974) and Cohen and Gold (1977) for the basic properties of these two (distinct) classes of ω-languages. A set $L \subseteq \Sigma^{\infty}$ is <u>∞-regular</u> (<u>deterministic ∞-regular</u>) if $L \cap \Sigma^*$ is a regular language and $L \cap \Sigma^{\omega}$ is an ω-regular (deterministic ω-regular) ω-language.

A map $g : \Sigma^{\infty} \to \Delta^{\infty}$ is a <u>proper homomorphism</u> if $g(\Sigma) \subseteq \Delta^*$, $g : \Sigma^* \to \Delta^*$ is a homomorphism and $g(a_0a_1a_2\ldots) = g(a_0)g(a_1)g(a_2)\ldots$ for every ω-word $a_0a_1a_2\ldots \in \Sigma^{\omega}$. For $g : \Sigma^{\infty} \to \Delta^{\infty}$, we write $|g| = \max\{|g(a)| \mid a \in \Sigma\}$.

A proper homomorphism $g : \Sigma^{\infty} \to \Delta^{\infty}$ is <u>effectively given</u> by listing $g(a)$ for all $a \in \Sigma$. An ω-regular language L is <u>effectively given</u> by giving a finite state automaton M such that $L = L^{\omega}(M)$.

When $\alpha,\gamma \in \Sigma^{\infty}$, we write $\alpha \leq \gamma$ to mean that α is a prefix of γ. When $v \in \Sigma^*$, we denote the string $vvv\ldots$ by v^{ω}.

For basic notions in formal languages see Salomaa (1973).

2. Deterministic ω-regular ω-languages

Here we state three simple results that will be needed later. The first lemma is probably known but we have found no reference.

<u>Lemma 1</u> Deterministic ω-regular ω-languages are effectively closed under intersection.

<u>Proof</u>: Let $M_i = (K_i, \Sigma, \delta_i, q_{i0}, F_i)$, $i = 1,2$, be two deterministic finite state automata. Define $M_3 = (K_3, \Sigma, \delta_3, q_{30}, F_3)$ by

$$K_3 = \{0,1,2\} \times K_1 \times K_2$$
$$F_3 = \{2\} \times K_1 \times K_2$$
$$q_{30} = (0, q_{10}, q_{20})$$
$$\delta_3((j, q_1, q_2), a) = (j', q_1', q_2')$$
$$\text{if } q_i' = \delta_i(q_i, a) \qquad \text{for } i = 1,2$$
$$\text{and } j' = \pi(j, q_1', q_2') \quad ,$$

where the function $\pi : \{0,1,2\} \times K_1 \times K_2 \to \{0,1,2\}$ is defined by

$$\begin{aligned}
\pi(2, p, q) &= 0 \\
\pi(0, p, q) &= 0 && \text{if } p \notin F_1 \\
&= 1 && \text{if } p \in F_1 \\
\pi(1, p, q) &= 1 && \text{if } q \notin F_2 \\
&= 2 && \text{if } q \in F_2
\end{aligned}$$

From the definition of π it follows that a run of M_3 on $\alpha \in \Sigma^\omega$ enters a state in F_3 infinitely many times if and only if both the corresponding runs of M_1 and M_2 on α enter final states infinitely many times.

Therefore $L^\omega(M_3) = L^\omega(M_1) \cap L^\omega(M_2)$.

□

Lemma 2 Deterministic ω-regular ω-languages are effectively closed under union.

Proof: See Eilenberg (1974).

□

Lemma 3 If $g : \Sigma^\infty \to \Delta^\infty$ is a proper homomorphism then $g^{-1}(\Delta^\omega) \cap \Sigma^\omega$ is a deterministic ω-regular set.

Proof: $g^{-1}(\Delta^\omega) \cap \Sigma^\omega = L^\omega(M)$ for this finite state automaton $M = (K, \Sigma, \delta, q_0, F)$: $K = \{q_0, q_1\}$, $F = \{q_0\}$,

$$\delta(q_i, a) = q_1 \quad \text{if} \quad g(a) = \varepsilon , \quad i = 1,2 ;$$

$$\delta(q_i, a) = q_0 \quad \text{if} \quad g(a) \neq \varepsilon , \quad i = 1,2 .$$

□

3. ω-equality Sets

Throughout this section we consider two fixed proper homomorphisms $g,h : \Sigma^\infty \to \Delta^\infty$; for s in Σ^* we define the balance of s by

$$\beta(s) = |g(s)| - |h(s)|$$

where $|w|$ denotes the length of $w \in \Delta^*$. Let $|\beta| = \max\{|\beta(a)| \mid a \in \Sigma\}$.

The ω-equality set is denoted by E^ω (or $E^\omega(g, h)$ if g,h are not understood) and defined as

$$E^{\omega}(g,\ h) = \{\alpha \in \Sigma^{\omega} \mid g(\alpha) = h(\alpha)\}\ .$$

For $k = 1,2,3,\ldots$ let

$$E_k^{\omega}(g,\ h) = E_k^{\omega} = \left\{\alpha \in E^{\omega}\ \middle|\ |\beta(s)| \le k \text{ for each } s \in \Sigma^*,\ s \le \alpha\right\} \cup \bigcup_{0<|v|\le k}\left\{t\gamma \in E^{\omega}\ \middle|\ t \in \Sigma^*,\ \gamma \in \Sigma^{\omega},\ |\beta(s)| \le k \text{ for each } s \le t, \text{ and } [g(\gamma) = v^{\omega} \text{ or } h(\gamma) = v^{\omega}]\right\}.$$

Clearly, $E_1^{\omega} \subseteq E_2^{\omega} \subseteq \ldots \subseteq E^{\omega}$.

Theorem 1   Each $E_k^{\omega}(g,\ h)$ is a deterministic ω-regular ω-language. Moreover, if k, g and h are effectively given then we can effectively give a deterministic finite state automaton M such that $E_k^{\omega}(g,\ h) = L^{\omega}(M)$.

Proof: For $v \in \Delta^*$, $0 < |v| \le k$, let $D(g,\ h,\ k,\ v)$ be the set of the $\alpha \in \Sigma^{\omega}$ with these two properties:

(1) for each finite $s \le \alpha$ we have

$$g(s) \le h(s) \quad \text{or} \quad h(s) \le g(s) \quad ;$$

(2) either (2a) $|\beta(s)| \le k$ for each finite $s \le \alpha$

or (2b) $\alpha = t\gamma$, $t \in \Sigma^*$, $\gamma \in \Sigma^{\omega}$, $|\beta(s)| \le k$ for each $s \le t$, and $g(s_1) \le v^{\omega}$, $h(ts_1) \le g(t)v^{\omega}$ for each finite $s_1 \le \gamma$.

We shall show that each $D(g,\ h,\ k,\ v)$ is (effectively) a deterministic ω-regular ω-language. Then E_k^{ω} is the union of the sets

$$\left\{\alpha \in \Sigma^{\omega} \;\middle|\; |\beta(s)| \leq k \text{ for each finite } s \leq \alpha \text{ and } g(s) = h(s) \text{ for infinitely many } s \leq \alpha\right\} ,$$

$$D(g, h, k, v) \cap g^{-1}(\Delta^{\omega}) \cap h^{-1}(\Delta^{\omega}) \quad , \quad 0 < |v| \leq k \quad , \text{ and}$$

$$D(h, g, k, v) \cap g^{-1}(\Delta^{\omega}) \cap h^{-1}(\Delta^{\omega}) \quad , \quad 0 < |v| \leq k \quad .$$

The first set is deterministic ω-regular (this follows from (Salomaa, 1978), Theorem 2.4); the remaining sets are deterministic ω-regular by Lemmas 1 and 3. Hence the (effective) existence of M follows from Lemma 2.

The deterministic finite state automaton M_1 accepting $D(g, h, k, v)$ works as follows: For an ω-word α on input, M_1 keeps comparing the values of g and h on finite prefixes s of α as long as $|\beta(s)| \leq k$. If and when $|\beta(s)|$ exceeds k , the tails of both $g(\alpha)$ and $h(\alpha)$ are matched agains v^{ω} .

Formally, M_1 has five kinds of states:

(i) the state $S(+, \varepsilon) = S(-, \varepsilon)$,

(ii) states $S(+, w)$ and $S(-, w)$ for each $w \in \Delta^*$, $0 < |w| \leq k$,

(iii) states $S(+, w, u)$ for $w \in \Delta^*$, $0 < |w| \leq k$, $u \leq v$,

(iv) states $S(u_g, u_h)$ for $u_g \leq v$, $u_h \leq v$.

(v) the state $S(*)$ (dead state).

All states of M_1 except $S(*)$ are final; the initial state is $S(+, \varepsilon) = S(-, \varepsilon)$.

The transition function δ is defined by

$$
\begin{array}{llll}
\delta(S(+,w),a) &= S(+,w') & \text{if} & wg(a) = h(a)w' \\
\delta(S(+,w),a) &= S(-,w') & \text{if} & wg(a)w' = h(a) \\
\delta(S(+,w),a) &= S(+,w',u) & \text{if} & |wg(a)| > |h(a)| + k \,,\ g(a) = v^i u \\
 & & & \text{for some } i \geq 0 \,,\ w = h(a)w' \,,\ w' \neq \varepsilon \\
\delta(S(+,w),a) &= S(u_g,u_h) & \text{if} & \big||wg(a)| - |h(a)|\big| > k \,, \\
 & & & g(a) = v^i u_g \quad \text{for some } i \geq 0 \,, \\
 & & & h(a) = wv^m u_h \quad \text{for some } m \geq 0 \\
\delta(S(-,w),a) &= S(+,w') & \text{if} & wh(a)w' = g(a) \\
\delta(S(-,w),a) &= S(-,w') & \text{if} & wh(a) = g(a)w' \\
\delta(S(-,w),a) &= S(u_g,u_h) & \text{if} & \big||wh(a)| - |g(a)|\big| > k \,, \\
 & & & g(a) = v^i u_g \quad \text{for some } i \geq 0 \,, \\
 & & & wh(a) = v^m u_h \quad \text{for some } m \geq 0 \\
\delta(S(+,w,u),a) &= S(+,w',u') & \text{if} & ug(a) = v^i u' \quad \text{for some } i \geq 0 \,, \\
 & & & w = h(a)w' \,,\ w' \neq \varepsilon \\
\delta(S(+,w,u),a) &= S(u_g,u_h) & \text{if} & ug(a) = v^i u_g \quad \text{for some } i \geq 0 \,, \\
 & & & h(a) = wv^m u_h \quad \text{for some } m \geq 0 \\
\delta(S(u_g,u_h),a) &= S(u_g',u_h') & \text{if} & u_g g(a) = v^i u_g' \quad \text{for some } i \geq 0 \,, \\
 & & & u_h h(a) = v^m u_h' \quad \text{for some } m \geq 0 \,.
\end{array}
$$

All other values of δ are $S(*)$.

It follows from the construction that for $x \in \Sigma^*$ we have

$$
\begin{aligned}
\delta(S(+,\varepsilon),x) &= S(+,w) \quad \text{iff } g(x)=h(x)w \text{ and } |\beta(s)| \le k \text{ for each } s \le x \\
&= S(-,w) \quad \text{iff } g(x)w=h(x) \text{ and } |\beta(s)| \le k \text{ for each } s \le x \\
&= S(+,w,u) \quad \text{iff there is } t<x,\ t \text{ is the longest prefix of } x \text{ such that } |\beta(s)| \le k \\
&\qquad\qquad \text{for each } s \le t,\ h(x)w = g(t) \text{ and } g(x) = g(t)v^i u \text{ for some } i \ge 0 \\
&= S(u_g, u_h) \quad \text{iff there is } t<x,\ t \text{ is the longest prefix of } x \text{ such that } |\beta(s)| \le k \\
&\qquad\qquad \text{for each } s \le t, \\
&\qquad\qquad g(x) = g(t)v^i u_g \text{ for some } i \ge 0, \text{ and} \\
&\qquad\qquad h(x) = g(t)v^m u_h \text{ for some } m \ge 0. \\
&= S(*) \quad \text{otherwise}
\end{aligned}
$$

Hence $D(g, h, k, v) = L^\omega(M_1)$.

□

Observe that, in terms of Cohen and Gold (1977), M also 1'-accepts $D(g, h, k, v)$; hence $D(g, h, k, v)$ is even a regular adherence set in the sense of Boasson and Nivat (1979).

Theorem 1 together with the following Theorem 2 enable us to deal with ω-regular subsets of E_k^ω effectively.

<u>Theorem 2</u> If $R \subseteq \Sigma^\omega$ is ω-regular and $R \subseteq E^\omega(g, h)$ then $R \subseteq E_k^\omega(g, h)$ for some k . More precisely, if $R = L^\omega(M)$ for a finite state automaton with n states then $R \subseteq E_k^\omega(g, h)$ where $k = n \cdot \max(1, |\beta|, |g|, |h|)$.

Proof: Take M such that $R = L^{\omega}(M)$ and M has n states; put $k = n\cdot\max(1, |\beta|, |g|, |h|)$. Choose any $\alpha \in R$. If $|\beta(s)| \leq k$ for each finite prefix s of α then $\alpha \in E_k^{\omega}$. Otherwise, there is $y \leq \alpha$ such that $|\beta(y)| > k$; take the shortest such y and write $\alpha = y\alpha_1$, $\alpha_1 \in \Sigma^{\omega}$. We have $|\beta|\cdot|y| \geq |\beta(y)| > k \geq n\cdot|\beta|$ and so $|y| > n$. It follows that $y = tuw$ with $0 < |u| \leq n$, $\beta(u) \neq 0$ and $tu^j w\alpha_1 \in R$ for each $j \geq 0$.

First assume $\beta(u) > 0$, i.e. $|g(u)| > |h(u)|$. Let $\gamma = uw\alpha_1$ and $v = g(u)$. If $h(u) = \varepsilon$ then $h(tw\alpha_1) = g(tu^j w\alpha_1)$ for all j , hence $g(\gamma) = v^w$ and therefore $\alpha \in E_k^{\omega}$. If $|h(u)| > 0$ then for every s in Σ^*, $s \leq \gamma$ there are $j \geq 0$ and $x \in \Sigma^*$ such that $g(tu^{2j}) = h(tu^{2j}s)x$, $|h(t)| \leq j|h(u)|$, $g(t) \leq h(tu^j)$ and $|g(u)|$ divides j . Hence $j|h(u)| = i|g(u)|$ for some i , and by Lemma 4 below (with $a_1 = h(t)$, $b_1 = h(u^j)$, $c_1 = h(s)x$, $a_2 = g(t)$, $b_2 = g(u^i)$, $c_2 = g(u^{2(j-i)})$) we get $h(ts) \leq g(tu^{2j})$. Therefore $g(t\gamma) = h(t\gamma) \leq g(tu^{\omega})$, and $g(\gamma) \leq g(u^{\omega})$. Thus either $g(\gamma) = v^{\omega}$ (hence $\alpha \in E_k^{\omega}$) or $g(\gamma) \in \Delta^*$. We show that the latter is impossible: if $g(\gamma) \in \Delta^*$ then there are $s \in \Sigma^*$ and $\alpha_2 \in \Sigma^{\omega}$ such that $\alpha = ts\alpha_2$, $g(ts) = h(ts)$ and $g(\alpha_2) = h(\alpha_2) = \varepsilon$. Since $tus\alpha_2 \in R$, we get $g(tus) = h(tus)$, in contradiction with
$|g(tus)| = |g(ts)| + |g(u)| > |h(ts)| + |h(u)| = |h(tus)|$.

Finally, if $\beta(u) < 0$, i.e. $|g(u)| < |h(u)|$, then the same argument shows that $h(uw\alpha_1) = h(u)^{\omega}$, hence $\alpha \in E_k^{\omega}$.

□

<u>Lemma 4</u> If $a_1b_1c_1a_2b_2c_2 \in \Sigma^*$, $a_1 \le a_2b_2$, $a_2 \le a_1b_1$, $a_1b_1^2c_1 = a_2b_2^2c_2$ and $|b_1| = |b_2|$ then $a_1c_1 = a_2c_2$.

<u>Proof</u>: Without loss of generality, assume $a_1 \le a_2$, i.e. $a_2 = a_1w$, $w \in \Sigma^*$. Since $a_1w \le a_1b_1$, we have $b_1 = wu$, $u \in \Sigma^*$. Next $a_1b_1 \le a_2b_2$, so $b_2 = uv$, and $|v| = |w|$. Now

$$a_1wuwuc_1 = a_1b_1^2c_1 = a_2b_2^2c_2 = a_1wuvb_2c_2 \quad ,$$

hence $v = w$. Since $a_1wuwuc_1 = a_1wuwuwc_2$, we get $c_1 = wc_2$ and $a_1c_1 = a_1wc_2 = a_2c_2$.

□

Corollaries to Theorems 1 and 2 :

<u>Corollary 1</u> If the set $E^\omega(g, h)$ is ω-regular then it is deterministic ω-regular.

□

<u>Corollary 2</u> For given finite state automaton M and two homomorphisms g and h , it is decidable whether or not $g(\alpha) = h(\alpha)$ for each $\alpha \in L^\omega(M)$.

<u>Proof</u>: By Theorem 2, $g(\alpha) = h(\alpha)$ for each α in $L^\omega(M)$ if and only if $L^\omega(M) \subseteq E_k^\omega$ where $k = n \cdot \max(1, |\beta|, |g|, |h|)$ and n is the number of states in M . By Theorem 1, $E_k^\omega = L^\omega(M_1)$ for some (deterministic) finite state automaton M_1 . The property $L^\omega(M) \subseteq L^\omega(M_1)$ is effectively testable by (Cohen and Gold, Th. 2.2.5).

□

4. Finite State Transducers

In this section we look at the infinite behavior of finite state transducers. We write a (nondeterministic) finite state transducer as $T = (K, \Sigma, \Delta, \delta, q_0, F)$, where K is a finite set of states, Σ is the input alphabet, Δ is the output alphabet, $\delta : K \times (\Sigma \cup \{\varepsilon\}) \to 2^{K\times\Delta^*}$ is the transition (and output) function, $q_0 \in K$ is the initial state and $F \subseteq K$ is the set of final states. If $\alpha \in \Sigma^\infty$, $\alpha = a_0a_1a_2\ldots$, $a_i \in \Sigma \cup \{\varepsilon\}$ for $i \geq 0$, $\gamma \in \Delta^\infty$, $\gamma = u_0u_1u_2\ldots$, $u_i \in \Delta^*$ for $i \geq 0$, and $r = p_0p_1p_2\cdots$ is an infinite sequence of states $p_i \in K$, we say that r is a <u>run of T on α with output</u> γ if $p_0 = q_0$ and $(p_i, u_{i-1}) \in \delta(p_{i-1}, a_{i-1})$ for $i \geq 1$; in symbols we write $\alpha \overset{r}{\to} \gamma$.

Now we will introduce two relations $R^\infty(T)$ and $R^\omega(T)$ defined by T :

$$R^\infty(T) = \left\{(\alpha, \gamma) \in \Sigma^\infty \times \Delta^\infty \mid \alpha \overset{r}{\to} \gamma \text{ for some run } r \text{ such that } \mathrm{INS}(r) \cap F \neq \phi\right\} ,$$

$$R^\omega(T) = R^\infty(T) \cap (\Sigma^\omega \times \Delta^\infty) .$$

The relation $R^\infty(T)$ describes the behavior of transducer T on both finite and infinite strings, the relation $R^\omega(T)$ on infinite strings only. Note, however, that the range of $R^\omega(T)$ might contain finite strings, because T might read infinite input and produce empty output.

The analogue of Nivat's factorization theorem (Eilenberg 1974, Theorem IX.2.2) takes this form:

Theorem 3 For a set $A \subseteq \Sigma^\infty \times \Delta^\infty$, two conditions are equivalent:

(i) $A = R^\infty(T)$ for a finite state transducer T;

(ii) there is an ω-regular set $B \subseteq \Gamma^\omega$ and two proper homomorphisms $g : \Gamma^\infty \to \Sigma^\infty$ and $h : \Gamma^\infty \to \Delta^\infty$ such that

$$A = \{(g(\gamma), h(\gamma)) \mid \gamma \in B\} \quad .$$

□

The following lemma is easy to verify.

Lemma 5 Let $g : \Sigma^\infty \to \Delta^\infty$ be a proper homomorphism. If $R \subseteq \Sigma^\infty$ is ∞-regular then $g(R) \subseteq \Delta^\infty$ is ∞-regular. If $R \subseteq \Delta^\infty$ is ∞-regular then $g^{-1}(R) \subseteq \Sigma^\infty$ is ∞-regular.

□

Theorem 4 If T is a finite state transducer, then the domain and the range of $R^\infty(T)$ are ∞-regular sets.

Proof: Follows by Theorem 3 and Lemma 5.

□

Moreover, the domain and the range of $R^\infty(T)$ can be effectively given when T is effectively given (because the constructions in Theorem 3 and Lemma 5 are effective); this fact will be used in the proof of Theorem 5.

A finite state transducer T is called functional if the relation $R^\infty(T) \subseteq \Sigma^\infty \times \Delta^\infty$ is a partial function from Σ^∞ to Δ^∞.

<u>Theorem 5</u> There is an algorithm to decide, for two given functional finite state transducers T_1 and T_2, whether $R^{\infty}(T_1) \subseteq R^{\infty}(T_2)$.

<u>Proof</u>: Let $T_i = (K_i, \Sigma, \Delta, \delta_i, q_{i0}, F_i)$, $i = 1,2$. Let $\overline{\Delta}$ be another copy of Δ, disjoint from Δ; the correspondence between Δ^* and $(\overline{\Delta})^*$ will be written as $x \mapsto \overline{x}$, $x \in \Delta^*$. Define two proper homomorphisms $g,h : (\Delta \cup \overline{\Delta})^{\infty} \to \Delta^{\infty}$ by $g(a) = a$, $g(\overline{a}) = h(a) = \varepsilon$ and $h(\overline{a}) = a$ for $a \in \Delta$.

We shall construct a finite state transducer T_3 such that $R^{\infty}(T_1) \subseteq R^{\infty}(T_2)$ if and only if

(i) the domain of $R^{\infty}(T_1)$ is contained in the domain of $R^{\infty}(T_2)$,

and (ii) $g(\alpha) = h(\alpha)$ for each α in the range of $R^{\infty}(T_3)$.

This together with Corollary 2 (in Section 3) and Theorem 4 establishes the result.

The transducer T_3 simulates the simultaneous operation of T_1 and T_2 and produces their two outputs intermixed (the outputs are then separated by g and h).

Formally, we define $T_3 = (K_3, \Sigma, \Delta\cup\overline{\Delta}, \delta_3, q_{30}, F_3)$ where

$$K_3 = \{0,1,2\} \times K_1 \times K_2$$
$$q_{30} = (0, q_{10}, q_{20})$$
$$F_3 = \{2\} \times K_1 \times K_2 \quad .$$

To define δ_3, first define two other functions δ_4 and δ_5; the formula for δ_4 employs the function $\pi : \{0,1,2\} \times K_1 \times K_2 \to \{0,1,2\}$ defined in the proof of Lemma 1.

Let

$$\delta_4((j, q_1, q_2), a) = \Big\{((\pi(j, q_1', q_2'), q_1', q_2'), x_1\overline{x}_2) \mid (q_i', x_i) \in \delta_i(q_i, a) \text{ for } i = 1,2\Big\} ,$$

$$a \in \Sigma \cup \{\varepsilon\} ;$$

$$\delta_5((j, q_1, q_2), a) = \phi \quad \text{for } a \in \Sigma ;$$

$$\delta_5((0, q_1, q_2), \varepsilon) = \Big\{((j, q_1', q_2), x_1) \mid (q_1', x_1) \in \delta_1(q_1, \varepsilon) \text{ and either } [q_1' \notin F_1 \text{ and } j = 0] \text{ or } [q_1' \in F_1 \text{ and } j = 1]\Big\}$$

$$\cup \Big\{((0, q_1, q_2'), \overline{x}_2) \mid (q_2', x_2) \in \delta_2(q_2, \varepsilon)\Big\} ;$$

$$\delta_5((1, q_1, q_2), \varepsilon) = \Big\{((1, q_1', q_2), x_1) \mid (q_1', x_1) \in \delta_1(q_1, \varepsilon)\Big\}$$

$$\cup \Big\{((j, q_1, q_2'), \overline{x}_2) \mid (q_2', x_2) \in \delta_2(q_2, \varepsilon) \text{ and either } [q_2' \notin F_2 \text{ and } j = 1] \text{ or } [q_2 \in F_2 \text{ and } j = 2]\Big\} ;$$

$$\delta_5((2, q_1, q_2), \varepsilon) = \Big\{((0, q_1', q_2), x_1) \mid (q_1', x_1) \in \delta_1(q_1, \varepsilon)\Big\}$$

$$\cup \Big\{((0, q_1, q_2'), \overline{x}_2) \mid (q_2', x_2) \in \delta_2(q_2, \varepsilon)\Big\} .$$

Finally, let

$$\delta_3(p, a) = \delta_4(p, a) \cup \delta_5(p, a) \quad \text{for } p \in K_3 , \quad a \in \Sigma \cup \{\varepsilon\} .$$

The definition of δ_3 and F_3 ensures that if $r = (j_0, q_{10}, q_{20})(j_1, q_{11}, q_{21})(j_2, q_{12}, q_{22}) \ldots$ is a run of T_3 on $\alpha \in \Sigma^\infty$ and r_1 and r_2 are the corresponding runs of T_1 and T_2 on α , then $INS(r) \cap F_3 \neq \phi$ iff $INS(r_1) \cap F_1 \neq \phi$ and $INS(r_2) \cap F_2 \neq \phi$. Therefore T_3 has the desired properties and the proof is completed.

□

Corollary 3 Given two functional finite transucers T_1 and T_2 , it is decidable whether $R^\infty(T_1) = R^\infty(T_2)$.

Proof: The decidability of inclusion implies the decidability of equality.

□

We say that two finite transducers T_1 and T_2 are ω-equivalent if $R^\omega(T_1) = R^\omega(T_2)$.

Corollary 4 The ω-equivalence problem for functional finite transducers is decidable.

Proof: Follows from Corollary 3 and the following lemma, which says that finite transducers on infinite strings are closed under the restriction to an ω-regular set.

□

Lemma 6 Let $T = (K, \Sigma, \Delta, \delta, q_0, F)$ be a finite transducer and $L \subseteq \Sigma^\omega$ an ω-regular set. Then we can construct a finite transducer T_1 such that $R^\infty(T_1) = R^\infty(T) \cap (L \times \Delta^\infty)$.

Proof: As in Theorem 3, find Γ , $g : \Gamma^\infty \to \Sigma^\infty$, $h : \Gamma^\infty \to \Delta^\infty$ and $B \subseteq \Gamma^\omega$ such that $R^\infty(T) = \{(g(\gamma), h(\gamma)) \mid \gamma \in B\}$. The set $g^{-1}(L) \subseteq \Gamma^\infty$ is ∞-regular by Lemma 5, and $B \cap g^{-1}(L) \subseteq \Gamma^\omega$ is ω-regular because ω-regular sets are closed under intersection (Eilenberg, 1974; Chapter XIV). Since $R^\infty(T) \cap (L \times \Delta^\infty) = \{(g(\gamma), h(\gamma)) \mid \gamma \in B \cap g^{-1}(L)\}$, the result follows by Theorem 3.

□

We conclude this section by showing that the ω-equivalence is undecidable for general finite transducers, even for gsm as defined in (Salomaa, 1973).

Theorem 6 The ω-equivalence problem for (nondeterministic) gsm is undecidable.

Proof: Consider any gsm $M = (K, \Sigma, \Delta, \delta, q_0, F)$. Modify it to M' so that $M' = (K \cup \{f\}, \Sigma \cup \{\#\}, \Delta \cup \{\#\}, \delta', q_0, \{f\})$ where $f \notin K$, $\# \notin \Sigma \cup \Delta$, and δ is extended to δ' by $(f, \#) \in \delta'(q, \#)$ for each $q \in F$, $(f, \#) \in \delta'(f, \#)$.

Clearly M_1 and M_2 are equivalent iff M_1' and M_2' , constructed as above, are ω-equivalent. That completes the proof since the equivalence for (nondeterministic) gsm is undecidable (Berstel, 1979).

□

References

Berstel, J. (1979), Transductions and context-free languages; B.G. Teubner, Stuttgart.

Boasson, L. and Nivat, M. (1979), Adherences of languages; Université Paris, Laboratoire Informatique Theorique et Programmation, Res. Rep. N° 79-13.

Cohen, R.S. and Gold, A.Y. (1977), Theory of ω-languages. I: Characterization of ω-context-free languages; J. Comput. System Sci. 15, 169-184.

Eilenberg, S. (1974), Automata, Languages and Machines, Volume A; Academic Press, New York.

Culik, K. II (1979), Some decidability results about regular and pushdown translations; Information Processing Letters 8, 5-8.

Culik, K. II and Salomaa, A. (1978), On the decidability of homomorphism equivalence for languages; J. Comput. System Sci. 17, 163-175.

Salomaa, A. (1973), Formal Languages; Academic Press, New York.

Salomaa, A. (1978), Equality sets for homomorphisms of free monoids, Acta Cybernetica 4, 127-139.