All correspondence to second author

SIMPLIFICATIONS OF HOMOMORPHISMS†

by

A. Ehrenfeucht*

and

G. Rozenberg**

#CU-CS-114-77 September, 1977

† This is a revised version of our paper presented at the "Theoretical Computer Science" conference in August 1977 in Waterloo, Canada.

* A. Ehrenfeucht, Department of Computer Science, University of Colorado at Boulder, Boulder, Colorado 80309 USA

**G. Rozenberg, Department of Mathematics, University of Antwerp, U.I.A., Wilrijk, Belgium

WORKING PAPER

Our life is frittered away by detail ... simplify, simplify.

WALDEN, Book 2.

ABSTRACT

The notion of a simplification of a homomorphism is introduced and investigated. Its usefulness is demonstrated in providing rather short proofs of the following results:

(i) Given an arbitrary homomorphism h and arbitrary words x,y it is decidable whether or not there exists an integer n such that $h^n(x) = h^n(y)$.

(ii) Given an arbitrary homomorphism h and arbitrary words x,y it is decidable whether or not there exists integers n and r such that $h^n(x) = h^r(y)$.

(iii) Given an arbitrary DOL system G and an arbitrary integer d it is decidable whether or not G is locally caternative of depth not larger than d.

(iv) The equivalence problem for elementary polynomially bounded DOL systems is decidable.

I. INTRODUCTION

The notion of a homomorphism on a free monoid is one of the basic notions of formal language theory. It is definitely a very central notion in the theory of L systems (see, e.g., [2] and [6]) and it is the basic tool to define the sequences of words (DOL sequences) in this theory. However there is no doubt that our knowledge about the properties of homomorphisms is very limited which is very well demonstrated by the fact that most questions about basic properties of DOL sequences remain without an answer. Even if a particular problem is solved then it mostly involves a rather complicated solution mostly based on "invented on line" (ad hoc) techniques. Hence looking for some systematic proof techniques to deal with homomorphisms is a central (and quite challenging) topic in the theory of L systems in particular and in the formal language theory in general.

In this paper we propose such a technique and demonstrate its use in providing rather simple solutions to a number of quite basic problems about DOL sequences. The underlying idea is the following. Very often one gets the feeling that a given homomorphism "involves too many letters". We show that this is indeed true whenever homomorphism is not injective. Then one can consider another homomorphism which is defined on an alphabet of smaller cardinality and which preserves essential properties of the original homomorphism. Such "simplications" turn out to be especially useful for proofs by induction on the number of letters involved in the considered alphabet. Also this leads us naturally to consider elementary homomorphisms i.e. homomorphisms that cannot be simplified any more. They form a proper subclass of the class of injective homomorphisms.

Since a homomorphism is a basic component in a DOL system we extend this concept to DOL systems and consider simplifications of DOL systems. It turns out that this step provides an additional insight into DOL systems and moreover provides a useful proof technique to deal with a number of problems concerning DOL sequences.

We assume the reader to be familiar with rudiments of formal language theory (see, e.g., [8]) and with the rudiments of the theory of DOL systems (see, e.g., [3]). Perhaps the only unusual notation that we use is alph x to denote the set of all letters occurring in the word x. We use the notation $G = \langle \Sigma, h, \omega \rangle$ for a DOL system and $E(G) = \omega_0, \omega_1, \ldots$ for its sequence.

II. SIMPLIFICATIONS OF HOMOMORPHISMS

In this section we will present certain conditions under which in considering a homomorphism one can shift the attention to a homomorphism which is simplified in the sense that it is defined on a smaller number of letters.

The following result is obvious.

Lemma 1. Let Σ be a finite alphabet with $\#\Sigma = m$ and let h be a homomorphism on Σ^*. If h is not injective on Σ, then there exists $t < m$ and words $u_1,\ldots,u_t$ such that for every a in Σ, $h(a) \in \{u_1,\ldots,u_t\}$.

Now we will show that the analogous result holds if h happens to "glue together" two words (not necessary letters!) over Σ.

Theorem 1. Let Σ be a finite alphabet with $\#\Sigma = m$ and let h be a homomorphism on Σ^*. If h is not injective on Σ^* then there exist $t < m$ and nonempty words $u_1,\ldots,u_t$ such that, for every a in Σ, $h(a) \in \{u_1,\ldots u_t\}^*$.

Proof

(i) If h is an erasing homomorphism then the result obviously holds (take $U = \{h(a): a \in \Sigma \text{ and } h(a) \neq \Lambda\}$, then indeed, for every a in Σ, $h(a) \in U^*$).

(ii) Let us assume that h is a Λ-free homomorphism.

Let CONTR be the set of all counterexamples to the statement of this result, that is CONTR is the set of all 4-tuples (Σ,w_1,w_2,h) where w_1,w_2 are words over an alphabet Σ of m letters, $w_1 \neq w_2$, $h(w_1) = h(w_2)$ and h is a Λ-free homomorphism on Σ^* such that there do not exist $t < m$ and words $u_1,\ldots,u_t$ such that, for every a in Σ, $h(a) \in \{u_1,\ldots,u_t\}^*$.

We will prove the lemma by demonstrating that CONTR is the empty set.

This in turn is proved by contradiction.

Thus let $(\Sigma,w_1,w_2,h) \in$ CONTR where $\Sigma = \{x_1,\ldots,x_m\}$. Let $\ell(\Sigma,w_1,w_2,h) = |h(w_1)|$.

1) $\ell(\Sigma,w_1,w_2,h) \neq 1$. Otherwise $|h(w_1)| = |h(w_2)| = 1$ and so $|w_1| = |w_2| = 1$. Consequently h is not injective on Σ and by Lemma 1 (Σ,w_1,w_2,h) is not in CONTR.

2) We will show now that if $\ell(\Sigma,w_1,w_2,h) = n \geq 2$ then CONTR must contain an element z with $\ell(z) < n$. However, by 1) this yields a contradiction.

So let $w_1 = X_{i_1}\ldots X_{i_k}, w_2 = X_{j_1}\ldots X_{j_r}, w_1 \neq w_2, h(w_1) = h(w_2)$ and $|h(w_1)| = |h(w_2)| = n \geq 2$. We have the following possibilities of interrelations between w_1 and w_2 (clearly, $k,r \geq 2$).

2.1) $X_{i_1} = X_{j_1}$.

Then if we set $\bar{w}_1 = X_{i_2}\ldots X_{i_k}$, $\bar{w}_2 = X_{j_2}\ldots X_{j_r}$ we have $\bar{w}_1 \neq w_2$ and $h(\bar{w}_1) = h(\bar{w}_2)$. Thus $(\Sigma,\bar{w}_1,\bar{w}_1,h)$ is an element of CONTR with $\ell(\Sigma,\bar{w}_1,\bar{w}_2,h) < n$.

Note that the case of $X_{i_1} \neq X_{j_1}$ and $h(X_{i_1}) = h(X_{j_1})$ cannot happen because then h is not injective on Σ and so, by Lemma 1, (Σ,w_1,w_2,h) is not a counter-example.

2.2) $X_{i_1} \neq X_{j_1}$ and $h(X_{i_1}) \neq h(X_{j_1})$.

Now either $h(X_{j_1})$ is a strict prefix of $h(X_{i_1})$ or $h(X_{i_1})$ is a strict prefix of $h(X_{j_1})$. Because these cases are symmetric let us assume that $h(X_{i_1}) = h(X_{j_1})z$ for a nonempty word z. Let Y be a new symbol, $\overline{\Sigma} = \Sigma\backslash\{X_{i_1}\} \cup \{Y\}$ and let $\bar{h}$ be a Λ-free homomorphism on $\overline{\Sigma}^*$ such that $\bar{h}(Y) = z$ and $\bar{h}(b) = h(b)$ for b in $\Sigma\backslash\{X_{i_1}\}$. (Note that $\#\Sigma = \#\overline{\Sigma} = m$).

Let f be a Λ-free homomorphism on Σ^* such that $f(X_{i_1}) = X_{j_1} Y$ and $f(X) = X$ for $X \neq X_{i_1}$. Let $\bar{w}_i = f(w_i)$ for i = 1,2. Now we have the following

(i) The first letters in $\bar{w}_1$ and $\bar{w}_2$ are the same. This holds because X_{j_1} is the first letter in both $\bar{w}_1$ and $\bar{w}_2$.

(ii) $\bar{w}_1 \neq \bar{w}_2$.

This holds because the second letter in $\bar{w}_1$ is Y while the second letter in $\bar{w}_2$ is an element of Σ.

(iii) $h(w_i) = \bar{h}(\bar{w}_i)$ for i = 1,2.

This is proved as follows.

Let $X_r \in \Sigma$.

If $r \neq i_1$ then $f(X_r) = X_r$ and $h(X_r) = \bar{h}(X_r) = \bar{h}(f(X_r))$. If $r = i_1$ then $f(X_r) = X_{j_1} Y$ and $h(X_r) = h(X_{j_1})z = h(X_{j_1})\bar{h}(Y) = \bar{h}(X_{j_1} Y) = \bar{h}(f(X_{i_1}))$. Hence for every X_r in Σ, $h(X_r) = \bar{h}f(X_r)$ and consequently, for i = 1,2, $h(w_i) = \bar{h}f(w_i) = \bar{h}(\bar{w}_i)$. This proves (iii).

(iv) From (i) through (iii) it follows that $(\bar{\Sigma},\bar{w}_1,\bar{w}_2,h)$ is an element of CONTR which falls into the case 2.1).

But 2) follows now from 2.1) and 2.2); and from 1) and 2) it follows that CONTR is the empty set. Hence the theorem holds also for Λ-free homomorphisms.

As a direct application of the previous result we get the following corollary.

Corollary 1. Let Σ,Δ be alphabets with $\#\Sigma=m$. Let h be a homomorphism from Σ^* into Δ^* which is not injective. Then there exist an alphabet Θ with $\#\Theta < m$ and homomorphisms $f : \Sigma^* \to \Theta^*, g : \Theta^* \to \Delta^*$ such that $h = gf$.

The above corollary leads to the following definition.

Definition 1. Let h be a homomorphism from Σ^* into Δ^*. We say that h is simplifiable if there exist an alphabet Θ with $\#\Theta < \#\Sigma$ and homomorphisms $f : \Sigma^* \to \Theta^*$, $g : \Theta^* \to \Delta^*$ such that $h = gf$. Otherwise h is called elementary.

Corollary 1 says that a non-injective homomorphism is not elementary. However one can have an injective homomorphism that is not elementary as shown by the following example.

Example 1. Let $\Sigma = \{a,b,c\}$ and h be a homomorphism from Σ^* into Σ^* defined by $h(a) = a$, $h(b) = bca$, $h(c) = bcaa$. Clearly h is injective on Σ^*. However if we take $\Theta = \{x,y\}$ and $f : \Sigma^* \to \Theta^*$, $g : \Theta^* \to \Sigma^*$ homomorphisms defined by $f(a) = x$, $f(b) = yx$, $f(c) = yxx$, $g(x) = a$, $g(y) = bc$ then indeed $h = gf$.

Both injectiveness and simplifiability are effective notions which is shown next.

Theorem 2. 1) It is decidable whether an arbitrary homomorphism is injective. 2) It is decidable whether an arbitrary homomorphism is elementary.

Proof.

1) This follows directly from the lemma below which we believe is of interest on its own.

Lemma 2. Given an arbitrary homomorphism h on Σ^* one can effectively construct a finite automaton A_h such that $L(A_h) = \{w \in \Sigma^* : (\exists z)_{\Sigma^*}(z \neq w \text{ and } h(z) = h(w))\}$.

Proof of Lemma 2.

Let

$H = \bigcup_{a \in \Sigma} h(a)$,

$Eq(h) = \{(a,b) \in \Sigma \times \Sigma : a \neq b \text{ and } h(a) = h(b)\}$, and

$Neq(h) = \{(a,b) \in \Sigma \times \Sigma : h(a) \neq h(b)\}$.

Let $B_h = (\Delta, Q, \delta, q_{in}, F)$ be a finite automaton where

$\Delta = (\Sigma \cup \{\Lambda\}) \times (\Sigma \cup \{\Lambda\})$,

$Q = \{q_{in}, [\Lambda]\} \cup$

$\cup \{-\alpha : \alpha \text{ is a nonempty prefix or a nonempty suffix of a word in } H\}$

$\cup \{+\alpha : \alpha \text{ is a nonempty prefix or a nonempty suffix of a word in } H\}$,

$F = \{[\Lambda]\}$

and δ is defined as follows:

0) for $(a,a) \in \Sigma \times \Sigma$, $\delta(q_{in}, (a,a)) = \{q_{in}\}$,

1) for $(a,b) \in Eq(h)$, $\delta(q_{in}, (a,b)) = \{q_{in}, [\Lambda]\}$,

2) for $(a,b) \in Neq(h)$

2.1) if $h(a) = h(b)\alpha$, then $\delta(q_{in}, (a,b)) = \{-\alpha\}$,

2.2) if $h(b) = h(a)\alpha$, then $\delta(q_{in}, (a,b)) = \{+\alpha\}$,

3) if $h(b)\beta = \alpha$, then $\delta(-\alpha, (\Lambda,b)) = \{-\beta\}$,

4) if $h(b) = \alpha\beta$, then $\delta(-\alpha, (\Lambda,b)) = \{+\beta\}$,

5) if $h(b) = \alpha$, then $\delta(-\alpha, (\Lambda,b)) = \{[\Lambda], q_{in}\}$,

6) if $h(a)\beta = \alpha$, then $\delta(+\alpha, (a,\Lambda)) = \{+\beta\}$,

7) if $h(a) = \alpha\beta$, then $\delta(+\alpha, (a,\Lambda)) = \{-\beta\}$,

8) if $h(a) = \alpha$, then $\delta(+\alpha, (a,\Lambda)) = \{[\Lambda], q_0\}$.

(As usual we assume that all nonspecified above transitions lead to a "dead state").

Now let ψ be a homomorphism from Δ^* to Σ^* such that, for every $(x,y) \in \Delta$, $\psi(x,y) = x$. Then obviously

$\psi(L(B_h)) = \{w \in \Sigma^* : (\exists z)_{\Sigma^*}(z \neq w \text{ and } h(z) = h(w))\}$.

But finite automata are (effectively) closed under homomorphic mappings and so the lemma holds.

Now the point 1) of the theorem follows from the well known fact that the emptiness for finite automata is decidable.

2) This is rather clear. Given a homomorphism from Σ^* into Σ^* it is enough to check for all the subalphabets Θ of Σ whether or not h "decomposes through Θ" in the sense of existence of homomorphisms $f : \Sigma^* \to \Theta^*$ and $g : \Theta^* \to \Sigma^*$, where g can be assumed Λ-free, such that $h = gf$; obviously this can be effectively checked.

Now in the rest of this paper we will demonstrate the usefulness of the concept of simplification by applying it to solve some rather basic problems concerning DOL systems. We assume in the sequel that DOL systems we deal with generate infinite languages, because otherwise the problems that we will consider become trivial.

III. APPLICATION 1 (Intersection of two orbits of the same homomorphism)

A quite natural problem in the theory of (iterated) homomorphisms is the following one: Let us start to iterate a homomorphism h on two different words. Do the sequences (orbits) generated by iterating h on these words ever meet?

We show that using the simplification mechanism one easily shows that the above problem is decidable.

Theorem 3. Let h be a homomorphism from Σ^* into Σ^* where $\#\Sigma = m$. Let w_1, w_2 be words over Σ. Then there exists an n such that $h^n(w_1) = h^n(w_2)$ if and only if $h^{m-1}(w_1) = h^{m-1}(w_2)$.

Proof.

By induction on m.

(i) m = 1 : obvious.

(ii) Assume that the result is true for $\#\Sigma \leq m-1$.

(iii) Let $\#\Sigma = m$.

If $w_1 = w_2$ then the result is obvious.

If $w_1 \neq w_2$ then h is not injective and so by Corollary 1 there exist an alphabet Θ with $\#\Theta < m$ and homomorphisms $f : \Sigma^* \to \Theta^*$, $g : \Theta^* \to \Sigma^*$ such that $h = gf$.

Then

$(\exists n)\ (h^n(w_1) = h^n(w_2))$ if and only if

$(fg)^n\ f(w_1) = (fg)^n\ f(w_2)$ if and only if (by the inductive assumption)

$(fg)^{m-2}\ f(w_1) = (fg)^{m-2}\ f(w_2)$ if and only if

$f\ h^{m-2}(w_1) = f\ h^{m-2}(w_2)$ which implies

$gf\ h^{m-2}(w_1) = gf\ h^{m-2}(w_2)$ if and only if

$h^{m-1}(w_1) = h^{m-1}(w_2)$.

Now as a direct application of the above result we get the following result.

Theorem 4. 1) There exists an algorithm which given an arbitrary homomorphism h on Σ and arbitrary words w_1, w_2 over Σ decides whether or not there exists an n such that $h^n(w_1) = h^n(w_2)$.

2) There exists an algorithm which given an arbitrary homomorphism h on Σ and arbitrary words w_1, w_2 over Σ decides whether or not there exist n and r such that $h^n(w_1) = h^r(w_2)$.

Proof.

1) This follows directly from Theorem 2.

2) First let us note that from Theorem 2 it follows that (we let $m = \#\Sigma$):

$(\exists n,r)(h^n(w_1) = h^r(w_2))$ if and only if

$(\exists \bar{n})(h^{\bar{n}}(w_1) = h^{m-1}(w_2))$ or $(\exists \bar{r})(h^{\bar{r}}(w_2) = h^{m-1}(w_1))$.

But given an arbitrary homomorphism h and arbitrary words x and y it is decidable whether or not y is reachable from x by iterating h. Hence 2) holds.

IV. APPLICATION 2 (Locally catenative DOL systems)

One of the rather important areas in the theory of DOL systems is the theory of locally catenative DOL systems (see, e.g., [3], [5] and [7]). This theory deals with such DOL systems which generate sequences in which from some moment on each string is composed by catenating (in a fixed order) a number of previous strings. In this section we will demonstrate the use of the simplification technique to solve one of the open problems in this area.

Let us start by recalling some definitions.

Definition 2. Let $\zeta = \omega_0,\omega_1,\ldots$ be an infinite sequence of words.

1) A shift of ζ is a sequence $\omega_i,\omega_{i+1},\ldots$ where $i \geq 0$.

2) Let $v = \langle i_1,\ldots,i_k\rangle$ with $k \geq 2$ be a vector of positive integers ($\max\{i_1,\ldots,i_k\}$ is called the depth of v). We say that ζ is v-locally catenative if there exists an s such that $\omega_r = \omega_{r-i_1}\ldots\omega_{r-i_k}$ for all $r > s$.

3) We say that ζ is locally catenative if there exists a vector v such that ζ is v-locally catenative. If the depth of v equals d then we also say that ζ is locally catenative of depth d.

The following observation will be needed in the sequel.

Lemma 3. Let ζ and ζ' be infinite sequences of nonempty words where words in ζ are over Σ and words in ζ' are over Σ'. Let h be a homomorphism from Σ^* into $(\Sigma')^*$ such that $h(\zeta)$ is a shift of ζ'. If ζ is v-locally catenative then so is ζ'.

Proof.

Let $\zeta = x_0,x_1,\ldots$, $\zeta' = y_0,y_1,\ldots$ and let $h(\zeta) = y_u,y_{u+1},\ldots$. Let $v = \langle i_1,\ldots,i_k\rangle$ and let m be such that $x_m = x_{m-i_1}\ldots x_{m-i_k}$. Then

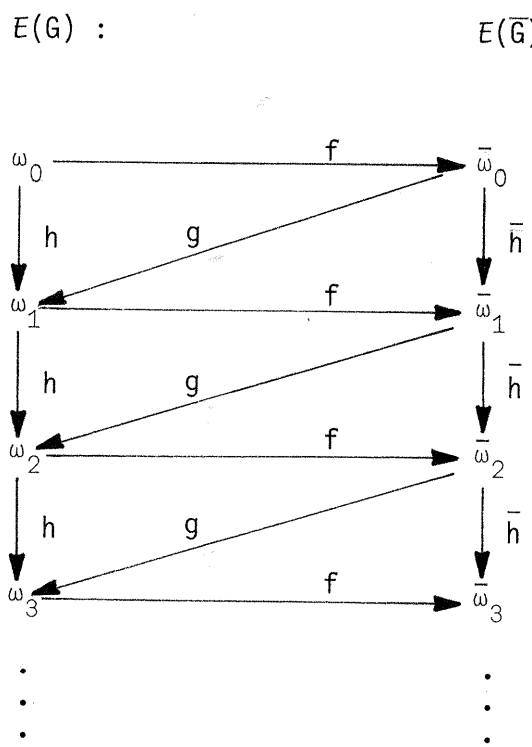
$y_{u+m} = h(x_m) = h(x_{m-i_1})\ldots h(x_{m-i_k}) = y_{u+m-i_1}\ldots y_{u+m-i_k}$. Consequently, ζ' is v-locally catenative.

Definition 3. A DOL system G is called (v-) locally catenative (of depth d) if $E(G)$ is (v'-) locally catenative (of depth d).

To investigate locally catenative DOL systems we shall extend now the notion of a simplification to DOL systems.

Definition 4. Let $G = \langle \Sigma, h, \omega \rangle$ and $\bar{G} = \langle \bar{\Sigma}, \bar{h}, \bar{\omega} \rangle$ be DOL systems. We say that $\bar{G}$ is a simplification of G if $\#\bar{\Sigma} < \#\Sigma$ and there exist homomorphisms $f : \Sigma^* \to \bar{\Sigma}^*$, $g : \bar{\Sigma}^* \to \Sigma^*$ such that $h = gf$, $\bar{h} = fg$ and $\bar{\omega} = f(\omega)$. If G has a simplification then it is called simplifiable, otherwise G is called elementary.

The following diagram illustrates the relationship between $E(G)$ and $E(\bar{G})$ in the case that $\bar{G}$ is a simplification of G.



Lemma 4. Let G, $\bar{G}$ be DOL systems such that $\bar{G}$ is a simplification of G. Then G is v-locally catenative if and only if $\bar{G}$ is v-locally catenative.

Proof.

It follows from Lemma 3 and from the observation that $f(E(G))$ is a shift of $E(\bar{G})$ and $g(E(\bar{G}))$ is a shift of $E(G)$, where f,g are as in Definition 4.

Lemma 5. Let G be an elementary DOL system and let $v = \langle i_1,\ldots,i_k\rangle$ be of depth d. Then G is v-locally catenative if and only if $\omega_d = \omega_{d-i_1}\cdots\omega_{d-i_k}$ where $E(G) = \omega_0,\omega_1,\ldots$.

Proof.

Let $G = \langle\Sigma,h,\omega\rangle$.

If $\omega_d = \omega_{d-i_1}\cdots\omega_{d-i_k}$ then obviously G is v-locally catenative.

If G is v-locally catenative then there exists the minimal n, say n_0, such that $\omega_{n_0} = \omega_{n_0-i_1}\cdots\omega_{n_0-i_k}$. If $n_0 > d$ then we arrive at contradiction as follows. We have $\omega_{n_0-1} \neq \omega_{n_0-1-i_1}\cdots\omega_{n_0-1-i_k}$ but

$$h(\omega_{n_0-1}) = \omega_{n_0} = \omega_{n_0-i_1}\cdots\omega_{n_0-i_k} = h(\omega_{n_0-1-i_1}\cdots\omega_{n_0-1-i_k})$$

and so, by Corollary 1, h is simplifiable; a contradiction.

One of the first natural questions posed when locally catenative DOL systems were introduced was whether it is decidable for an arbitrary DOL system G and a vector v whether G is v-locally catenative. The following result provides the affirmative answer to even more general question: is it decidable whether an arbitrary DOL system is locally catenative of depth no larger than a given positive integer?

Theorem 5. It is decidable for an arbitrary positive integer d and for an arbitrary DOL system G whether or not G is locally caternative of depth no greater than d.

Proof.

Let $G = \langle \Sigma, h, \omega \rangle$ with $E(G) = \omega_0, \omega_1, \ldots$ and $\#\Sigma = m$.

We claim that G is locally caternative of depth d if and only if ω_{d+m-1} is a catenation of some previous strings. Clearly it suffices to prove the only if part of this statement. But this follows from the fact that to find a simplification of G which is simple we have to make at most (m-1) consecutive simplifications of G, using pairs of homomorphisms $(f_1, g_1), \ldots, (f_t, g_t)$ with $t \leq m-1$. Then in the resulting system its d'th element must be a catenation of some previous strings (see Lemma 4). However then this sequence is shifted by a homomorphism $g_1 \cdots g_{t-1} g_t$ into $E(G)$ in such a way that the i'th element of this sequence is mapped into (i+t)'th element of $E(G)$. Hence, by Lemma 3, ω_{d+t} and consequently $\omega_{d+(m-1)}$ must be locally catenative.

V. APPLICATION 3 (DOL equivalence problem)

The (sequence) equivalence problem for DOL systems reads as follows: is it decidable whether or not two arbitrary DOL systems generate the same sequences. Although simply stated it turned out to be a challenging problem opened for several years; only recently it was shown in [1] that the problem is indeed decidable. However the proof turned out to be really complicated. As a matter of fact even for a simpler case of polynomially bounded DOL systems (see [4]) the proof of decidability of DOL equivalence problem is quite complicated. Hence it is still a challenging research topic to look for a simple proof of decidability of DOL equivalence problem for the whole class of DOL systems or for its nontrivial subclasses.

In this section we show how using the concept of simplification one can give a short and rather elegant proof that the equivalence for simple polynomially bounded DOL systems is decidable.

First we need some extra notation. Given two DOL systems $G_1 = \langle \Sigma, h_1, \omega \rangle$, $G_2 = \langle \Sigma, h_2, \omega \rangle$ with $E(G_1) = \omega_0^{(1)}, \omega_1^{(1)}, \ldots$ and $E(G_2) = \omega_0^{(1)}, \omega_1^{(1)}, \ldots$ let ϕ_{G_1,G_2}, ψ_{G_1,G_2} and ξ_{G_1,G_2} (or simply ϕ,ψ,ξ when G_1,G_2 are understood) be three set-valued sequences defined as follows:

for every $n \geq 0$,

$\phi(n) = \{\omega_n^{(1)}\}$,

$\psi(n) = \{\omega_n^{(2)}\}$ and

$\xi(n) = \{h_{i_n} \ldots h_{i_1} : i_1,\ldots,i_n \in \{1,2\}\}$.

Now let us describe an algorithm, called the ESP algorithm, which forms the basis for our next result.

INPUT: G_1, G_2

PROCEDURE: Construct, starting with n = 0 and then increasing n iteratively by 1 $\phi(n)$, $\psi(n)$, $\xi(n)$ (thus getting the sequence $\phi(0)$, $\psi(0)$, $\xi(0)$, $\phi(1)$, $\psi(1)$, $\xi(1)$,...).

STOP CONDITION: Stop at n such that

either 1) $\phi(n) \neq \psi(n)$,

or 2) the condition 1) does not hold and there exist two <u>different</u> sequences $i_n \ldots i_1$, $j_n \ldots j_1$ such that $h_{i_n} \ldots h_{i_1} = h_{j_n} \ldots h_{j_1}$.

<u>Lemma 6</u>. Given arbitrary two elementary polynomially bounded DOL systems, the ESP algorithm always terminate.

<u>Proof</u>.

Let $G_1 = \langle \Sigma, h_1, \omega \rangle$ and $G_2 = \langle \Sigma, h_2, \omega \rangle$ with $E(G_1) = \omega_0^{(1)}, \omega_1^{(1)}, \omega_2^{(1)}, \ldots$ and $E(G_2) = \omega_0^{(2)}, \omega_1^{(2)}, \omega_2^{(2)}, \ldots$ Let k be such that <u>alph</u> $(\omega_0^{(1)} \ldots \omega_k^{(1)}) = \Sigma$ (clearly without the loss of generality we can assume that such a k exists). Assume that $n \geq 1$ and that for $0 \leq \ell \leq k+n$ we have $\phi(\ell) = \psi(\ell)$. Consider $z = \omega_0^{(1)} \omega_1^{(1)} \ldots \omega_k^{(1)}$. Then clearly, for every $i_1, \ldots, i_n, j_1, \ldots, j_n \in \{1,2\}$, $h_{i_n} \ldots h_{i_1}(z) = h_{j_n} \ldots h_{j_1}(z)$ (call it z_n). The number of all possible parsings of z_n with respect to z does not exceed $|z_n|^{|z|}$ and, if P is a polynomial bounding the growth of G_1, this does not exceed $\left(\bigcup_{i=n}^{n+k} P(i)\right)^{|z|}$ which is again a polynomial. However the number of possible sequences of length n from the set {1,2} is 2^n and so there exists an n such that $2^n > \left(\bigcup_{i=n}^{n+k} P(i)\right)^{|z|}$. If such a minimal n is n_0 then the ESP algorithm terminates by the second termination condition on the n_0^{th} step.

Theorem 6. The equivalence problem for simple polynomially bounded DOL sequences is decidable. Moreover given two simple polynomially bounded DOL systems G_1, G_2 one can effectively calculate a constant C_{G_1,G_2} such that $E(G_1)$ agrees with $E(G_2)$ up to the C'_{G_1,G_2} th word if and only if $E(G_1) = E(G_2)$.

Proof.

In view of Lemma 6 it suffices to prove that if the ESP algorithm terminates by the second condition then $E(G_1) = E(G_2)$. Thus let us assume that the ESP algorithm terminates by the second condition on the n'th step i.e. there exist two different sequences $j_1,\ldots,j_n$, $i_1,\ldots i_n$ such that $h_{i_n}\ldots h_{i_1} = h_{j_n}\ldots h_{j_1}$. Let g be such that $i_n = j_n$, $i_{n-1} = j_{n-1},\ldots, i_{q+1} = j_{q+1}$ and $i_q \neq j_q$. Let $\tau = h_{i_n}\ldots h_{i_{q+1}} = h_{j_n}\ldots h_{j_{q+1}}$, $\tau_1 = h_{i_q}\ldots h_{i_1}$ and $\tau_2 = h_{j_q}\ldots h_{j_1}$.

Let us assume to the contrary that $E(G_1) \neq E(G_2)$. Let m be the minimal integer such that $\omega_m^{(1)} \neq \omega_m^{(2)}$. Then obviously $\omega_{m-q}^{(1)} = \omega_{m-q}^{(2)}$ and $\tau_1(\omega_{m-q}^{(1)}) = \tau_2(\omega_{m-q}^{(2)})$ (because before we reach ω_m each application of h_1 is identical with the application of h_2). However $\tau\tau_1 = \tau\tau_2$ and so $\tau\tau_1(\omega_{m-q}^{(1)}) = \tau\tau_2(\omega_{m-q}^{(2)})$. On the other hand $\omega_m^{(1)} \neq \omega_m^{(2)}$ but $\tau\tau_1(\omega_{m-q}^{(1)}) = \tau\tau(\omega_m^{(1)})$ and $\tau\tau_2(\omega_{m-q}^{(2)}) = \tau(\omega_m^{(2)})$ which yields $\tau(\omega_m^{(1)}) = \tau(\omega_m^{(2)})$. But this will imply that one of the elementary homomorphisms h_1, h_2 must be non-injective which contradicts Corollary 1.

Thus the algorithm to decide the equivalence of two simple polynomially bounded DOL systems G_1 and G_2 is as follows:

1. Run the ESP algorithm for G_1, G_2.
2. If it terminates by condition 1 then $E(G_1) = E(G_2)$.

3. If it terminates by condition 2 then $E(G_1) = E(G_2)$.

The termination of this algorithm is guaranteed by Lemma 4.

The second part of the theorem follows from the proof of Lemma 4 and the fact (see [2]) that for a given polynomially bounded DOL system one can effectively construct a polynomial P which bounds its growth. Thus to find a constant C_{G_1,G_2} it suffices to find an n satisfying the inequality $2^n > R(n)$ where R is a given polynomial. Clearly, this can be effectively done.

REFERENCES

[1] Culik II, K. and Fris. I., 1976, The decidability of the equivalence problem for DOL systems, Information and Control, to appear.

[2] Ehrenfeucht, A. and Rozenberg, G., 1976, On the structure of polynomially bounded DOL systems, Fundamenta Informatica, to appear.

[3] Herman, G.T. and Rozenberg, G., 1975, Developmental Systems and Languages, North Holland Publishing Company, Amsterdam.

[4] Karhumahi, J., 1976, The decidability of the equivalence problem for polynomially bounded DOL sequences, Revue Francaise d'Automatique, Informatique et Recherche Operationnelle, Ser. Rouge, to appear.

[5] Rozenberg, G. and Lindenmayer, A., 1973, Developmental systems with locally caternative formulas, Acta Informatica, 2:214-248.

[6] Rozenberg, G. and Salomaa, A., 1976, The mathematical theory of L systems, In: J. Tou (ed.), Advances in Information Systems Science, vol. 6, Plenum Press, New York, 161-206.

[7] Ruohonen, K., 1977, Remarks on locally caternative developmental sequences, submitted for publication.

[8] Salomaa, A., Formal languages, Academic Press, 1974.

ACKNOWLEDGMENTS

The authors greatfully acknowledge the financial support by the NSF Grant Number MCS75-09972 and by IBM Belgium. They are also very much indebted to L. Fosdick for making the work on this and other papers possible.