INFORMATION PROCESSING LETTERS 1 (1972) 201 - 204. NORTH-HOLLAND PUBLISHING COMPANY

THE EQUIVALENCE PROBLEM FOR DETERMINISTIC T0L-SYSTEMS IS UNDECIDABLE

Grzegorg ROZENBERG
Department of Computer Science, State University of New York at Buffalo, 4226 Ridge Lea Road, Amherst, New York 14226, U.S.A.
and
Mathematical Institute, Utrecht State University, Utrecht – Uithof, Netherlands

Received 17 April 1972

1. Introduction

L-languages were introduced by Lindenmayer [3] for the description of the development of filamentous organisms. Originally they were described in terms of linear arrays of automata, but in later works the formalism was changed to the more linguistical notion of an L-system. This gave rise to various families of developmental languages with an already fairly developed literature (see, e.g., ref. [4]).

This paper continues research on T0L-systems and languages introduced in ref. [6].

A T0L-system has the following components:
(*i*) A finite set of symbols, Σ the *alphabet*;
(*ii*) A starting string, ω, the *axiom*;
(*iii*) A finite collection P of *tables*, each of which is a finite set of *productions* which tell us by what strings in Σ^* a symbol may be replaced. A table may, in general, contain several productions allowing to replace the same symbol. In every step of a derivation, all symbols in a string must be simultaneously replaced according to the production rules, all of which are chosen from one (but arbitrary) table.

The language generated by the given T0L-system consists of all strings which can be derived from ω in a finite number of steps.

One of the important problems, from both, the biological and formal language theory points of view, is the equivalence problem for T0L-systems, which can be phrased as follows: Does there exist an algorithm, which, given arbitrary T0L-systems G_1 and G_2, will decide in a finite number of steps whether or not the languages generated by G_1 and G_2 are identical?

In this note we prove that such an algorithm cannot exist even for deterministic T0L-systems, i.e., such T0L-systems, where each table provides exactly one possibility of rewriting for every letter in the alphabet of the given system.

Recently such a result was obtained by Meera Blattner [7] for 0L-systems, i.e., such T0L-systems which contain only one table. The result presented here is however independent in the following sense:
(*i*) It was proved in ref. [6] (1972), that the classes of 0L and deterministic T0L languages are incomparable but not disjoint.
(*ii*) We prove our result by a direct construction (based on a simulation of the post correspondence problem).

2 Definitions

For the sake of completeness, we shall repeat (after ref. [6]) definitions of T0L-systems, languages, etc., as well as the post correspondence theorem (after ref. [1]). Our notation for basic terms from a formal language theory will follow this of ref. [1].

Definition 1

A *TOL-system* (over an alphabet Σ) is defined as a construct $G = \langle \Sigma, P, \sigma \rangle$, where

1. Σ is a finite set (called the alphabet of G).
2. P is a finite set, $P = \{T_1, ..., T_f\}$ for some $f \geqslant 1$, each element of which is a finite subset of $\Sigma \times \Sigma^*$. P satisfies the following (*completeness*) condition:

$$(\forall T)_P (\forall a)_\Sigma (\exists \alpha)_{\Sigma^*} (\langle a, \alpha \rangle \in T).$$

P is called *the set of tables* of G.

3. $\sigma \in \Sigma^+$ (called the axiom of G).

We assume that both Σ and P are non-empty sets.

Definition 2

Let $G = \langle \Sigma, P, \sigma \rangle$ be a TOL-system. Let

$$x \in \Sigma^+, x = a_1 ... a_k,$$

where each a_j, $1 \leqslant j \leqslant k$, is an element of Σ, and let $y \in \Sigma^*$. We say that x *directly derives* y in $G (x \Rightarrow_G y)$ if, and only if, there exist T in P and $p_1, ..., p_k$ in T such that

$$p_1 = \langle a_1, \alpha_1 \rangle, p_2 = \langle a_2, \alpha_2 \rangle, ..., p_k = \langle a_k, \alpha_k \rangle$$

(for some $\alpha_1, ..., \alpha_k \in \Sigma^*$) and $y = \alpha_1 ... \alpha_k$. We say that x derives y in G ($x \Rightarrow_G * y$) if, and only if, either

(*i*) there exists a sequence of words $x_0, x_1, ..., x_n$ in $\Sigma^* (n \geqslant 1)$ such that $x_0 = x$, $x_n = y$ and $x_0 \Rightarrow_G x_1 \Rightarrow_G \cdots \Rightarrow_G x_n$, or

(*ii*) $x = y$.

$\Rightarrow_G *$ simply denotes the transitive and reflexive closure of the relation $\Rightarrow_G$. $\Rightarrow_G^+$ shall denote the transitive closure of $\Rightarrow_G$.

Definition 3

Let $G = \langle \Sigma, P, \sigma \rangle$ be a TOL-system. The *language of* G (denoted as $L(G)$) is defined as

$$L(G) = \{x : 0 \Rightarrow_G * x\}.$$

Definition 4

Let Σ be a finite alphabet and $L \subseteq \Sigma^*$. L is called a *TOL language* if, and only if, there exists a TOL grammar G such that $L(G) = L$.

Definition 5

A TOL-system $G = \langle \Sigma, P, \sigma \rangle$ is called *deterministic* (denoted as DTOL) if, and only if, for each T in P and each a in Σ there exists exactly one element $\langle A, \alpha \rangle$ in T such that $A = a$.

Notation

Let $G = \langle \Sigma, P, \sigma \rangle$ be a TOL system. If $\langle a, \alpha \rangle$ is an element of some T in P, then we call it *a production* (for a in T) and write $a \to \alpha \in T$, or $a \to_T \alpha$. If $x \Rightarrow_G y$ "using" a table T of P, then sometimes we shall write $x \Rightarrow_T Y$.

Theorem 1 (Post correspondence theorem)

Let Σ contain at least two elements. Then it is recursively unsolvable to determine for arbitrary n-tuples $(\alpha_1, ..., \alpha_n)$ and $(\beta_1, ..., \beta_n)$ of nonempty words over Σ, whether there exists a (nonempty) sequence of indices $i_1, ..., i_k$ such that $\alpha_{i_1} \cdots \alpha_{i_k} = \beta_{i_1} \cdots \beta_{i_k}$.

The reader is referred to ref. [5] or [2] for the proof of this theorem.

3. Result

Theorem 2

It is recursively undecidable to determine for arbitrary DTOL-systems G_1, G_2 whether $L(G_1) = L(G_2)$.

Proof (outline):

Let Σ be a finite alphabet such that $\#\Sigma \geqslant 2$, say $\Sigma = \{a_1, ..., a_n\}$ for some $n \geqslant 2$. Let L_1, L_2 be an instance of the post correspondence problem over an alphabet Σ, say $L_1 = \langle \alpha_1, ..., \alpha_m \rangle$, $L_2 = \langle \beta_1, ..., \beta_m \rangle$ for some $m \geqslant 2$. Let $G_1 = \langle V, P_1, \omega \rangle$ be a DTOL system where

I. $V = \Sigma \cup \{S_1, S_2, R_1, R_2, U_1, U_2, A_1, A_2\} \cup \{1, ..., m\}$, where we assume that

$$\Sigma \cap (\{S_1, ..., A_2\} \cup \{1, ..., m\}) = \emptyset.$$

II. $\omega = S_1 S_2$.

III. P_1 consists of the following tables (which, for convenience are listed in groups):

0) $T_0 = \{S_1 \to A_1, S_2 \to A_2, X \to X$ for every X in $V - \{S_1, S_2\}\}$.

1) For every $1 \leqslant i \leqslant m$, $1 \leqslant k, l \leqslant n$, such that $k \neq l$, let

$$T_{1,i,k,l} = \{S_1 \to iR_1a_k, S_2 \to iR_2a_l, X \to X$$

for every X in $V - \{S_1, S_2\}\}$.

2) For every $1 \leqslant i \leqslant m$, $1 \leqslant j \leqslant n$, let

$$T_{2,i,j} = \{S_1 \to iU_1a_j, S_2 \to iU_2a_j, X \to X$$

for every X in $V - \{S_1, S_2\}\}$.

3) For every $1 \leqslant i \leqslant m$, $1 \leqslant j \leqslant n$, let

$$T_{3,i,j} = \{U_1 \to iU_1a_j, U_2 \to iU_2a_j, X \to X$$

for every X in $V - \{U_1, U_2\}\}$.

$$T_{3,0,j} = \{U_1 \to U_1a_j, U_2 \to U_2a_j, X \to X$$

for every X in $V - \{U_1, U_2\}\}$.

4) For every $1 \leqslant i \leqslant m$, $1 \leqslant k, l \leqslant n$, such that $k \neq l$, let

$$T_{4,0,k,l} = \{U_1 \to R_1a_k, U_2 \to R_2a_l, X \to X$$

for every X in $V - \{U_1, U_2\}\}$.

$$T_{4,i,k,l} = \{U_1 \to iR_1a_k, U_2 \to iR_2a_l, X \to X$$

for every X in $V - \{U_1, U_2\}\}$.

5) For every i, j, k, l such that $i, j \in \{0, ..., m\}$, $k, l \in \{0, ..., n\}$, let

$$T_{5,j,k,l} = \{R_1 \to t_jR_1v_k, R_2 \to t_iR_2v_l, X \to X$$

for every X in $V - \{R_1, R_2\}\}$,

where $t_0 = v_0 = \Lambda$, $t_i = i$ for $1 \leqslant i \leqslant m$, $v_j = a_j$ for $1 \leqslant j \leqslant n$.

6) $T_6 = \{R_1 \to A_1, R_2 \to A_2, X \to X$ for every X in $V - \{R_1, R_2\}\}$.

The following result is a direct consequence from the construction of G_1, and its (rather long but straightforward) formal proof is left to the reader.

Lemma 1

$L(G_1) \cap (\Sigma \cup \{1, ..., n\} \cup \{A_1, A_2\})^* =$
$= A_1A_2 \cup \{xA_1yxA_2\bar{y} : x \in \{1, ..., m\}^+, y, \bar{y} \in \Sigma^+, y \neq \bar{y}\}$. Let for every $1 \leqslant i \leqslant m$,

$$T_{7,i} = \{A_1 \to iA_1\alpha_i, A_2 \to iA_2\beta_i, X \to X$$

for every X in $V - \{A_1, A_2\}\}$.

Let $G_2 = \langle \Sigma, P_2, S_1S_2 \rangle$ be a DT0L-system such that

$$P_2 = P_1 \cup \{T_{7,i} : 1 \leqslant i \leqslant m\}.$$

Lemma 2

$L(G_1) = L(G_2)$ if, and only if, there does not exist a solution for the L_1, L_2 instance of the post correspondence problem.

Proof:

(*i*) Let us assume that $L(G_1) = L(G_2)$. Then from lemma 1, it follows that for every word (in $L(G_1)$) of the form $xA_1yxA_2\bar{y}$, where $x \in \{1, ..., m\}, y, \bar{y} \in \Sigma^+$ we have $y \neq \bar{y}$. But, from the construction of tables $T_{7,i}$ $(1 \leqslant i \leqslant m)$, it follows that for every sequence $i_1, ..., i_f$, $(f \geqslant 1)$, of elements from $\{1, ..., m\}$,

$$i_1 \cdots i_f A_1 \alpha_{i_f} \cdots \alpha_{i_1} i_1 \cdots i_f A \beta_f \cdots \beta_1$$

is in $L(G_2)$. Hence for every sequence $i_1, ..., i_f$, $(f \geqslant 1)$, of elements from $\{1, ..., m\}$, $\alpha_{i_f} \cdots \alpha_{i_1} \neq \beta_{i_f} \cdots \beta_{i_1}$, and so the post correspondence problem has no solution for the instance L_1, L_2.

(*ii*) Let us assume that the post correspondence problem has no solution for the instance L_1, L_2, which means that for every sequence $i_1, ..., i_f$, $(f \geqslant 1)$, of elements from $\{1, ..., m\}$, $\alpha_{i_1} \cdots \alpha_{i_f} \neq \beta_{i_1} \cdots \beta_{i_f}$.
But, from the construction of $T_{7,i}$ (for $1 \leqslant i \leqslant m$) it follows that

$$L(G_2) - L(G_1) = \{i_1 \cdots i_p A_1 \alpha_{i_p} \cdots \alpha_{i_1} i_1 \cdots i_p A_2 \beta_{i_p} \cdots \beta_{i_1} :$$

$$\text{where } p \geqslant 1, i_1, ..., i_p \in \{1, ..., m\}\},$$

and so (see lemma 1), $L(G_2) - L(G_1) \subseteq L(G_1)$, hence $L(G_1) = L(G_2)$.

From (*i*) and (*ii*) lemma 2 follows.

From lemma 2, we see, that, if the equivalence problem for DT0L-systems is recursively solvable, then the post correspondence problem is also recursively solvable, this is a contradiction. Thus, theorem 1 holds.

Corollary

It is recursively unsolvable to determine for arbitrary T0L-systems G_1, G_2 whether $L(G_1) = L(G_2)$.

Proof:

This follows from definitions of T0L and DT0L-systems (each DT0L-system is also a T0L-system).

Acknowledgement

The author is indebted to the referee for useful comments. The research for this paper has been supported by NSF Grant GJ998.

References

[1] S. Ginsburg, The mathematical theory of context free languages (McGraw-Hill, New York, 1966).
[2] J.E. Hopcroft, J.D. Ullman, Formal languages and their relation to automata (Addison–Wesley, Reading, Massachusetts, 1969).
[3] A. Lindenmayer, Mathematical models for cellular interactions in development, Parts I and II, J. Theoret. Biol., 18 (1968) 280.
[4] A. Lindenmayer, G. Rozenberg, Developmental systems and languages, in: Proc. ACM Symposium on Theory of Computing (1972).
[5] E.L. Post, A variant of a recursively unsolvable problem, Bull. Am. Math. Soc. 52 (1946) 264.
[6] G. Rozenberg, T0L systems and languages, Information and Control to appear (1972).
[7] S. Greibach, private communication.