The Unsolvability of the Equivalence Problem for Λ-Free Nondeterministic Generalized Machines

T. V. GRIFFITHS

Air Force Cambridge Research Laboratories, Bedford, Massachusetts.

ABSTRACT. It is shown that the equivalence problem for Λ-free nondeterministic generalized machines is unsolvable, and it is observed that this result implies the unsolvability of the equality problem for c-finite languages.

KEY WORDS AND PHRASES: Λ-free machines, nondeterministic machines, generalized machines, linear languages, c-finite languages, equivalence, unsolvability, language equality, Post correspondence problem, context-free languages, automata

CR CATEGORIES: 5.21, 5.22

Introduction

By showing that solvability of the equivalence problem for Λ-free (not having the null word Λ as output) nondeterministic generalized machines would imply solvability of Post's correspondence problem [2], we give negative answers to the following questions:

(1) Is the equivalence problem for nondeterministic generalized machines solvable? (Greibach)

(2) Is the equality problem for c-finite languages [1] solvable? (Ginsburg)

The result is perhaps a little surprising in view of the solvability of the equivalence problem for both deterministic generalized machines and nondeterministic complete machines. If M is a member of either of these classes, however, the output set that M produces in response to any given input string never has a member that is a proper prefix of any other member. It is easy to decide whether any two nondeterministic generalized machines having this property are equivalent, but from this fact little can be inferred about the more general problem.

We assume that Σ and Δ are finite nonempty sets, that $\Sigma^+ (\Delta^+)$ denotes the free semigroup over $\Sigma\ (\Delta)$ without identity, that $P(\Delta^+)$ denotes the set of finite subsets of Δ^+, that $|x|$ denotes that length of a string x in $\Sigma^+ \cup \Delta^+$, and that $x^{\#}$ denotes the reversal of x.

Definition 1. A *Λ-free nondeterministic generalized machine* (LFNGM) over (Σ, Δ) is a triple (K, M, q_0) where:

(i) K is a finite nonempty set (of states),

(ii) M is a function from $K \times \Sigma \times K$ into $P(\Delta^+)$ (the transduction function),

(iii) $q_0 \in K$ (the start state).

M is extended to $K \times \Sigma^+ \times K$ as follows. If $x, y \in \Sigma^+$,

$$M(p, xy, q) = \bigcup_{r \in K} M(p, x, r)M(r, y, q),$$

where $XY = \{xy: x \in X \ \&\ y \in Y\}$. The transduction from Σ^+ to $P(\Delta^+)$ defined by (K, M, q_0) is

$$M_T(x) = \bigcup_{q \in K} M(q_0, x, q).$$

Definition 2. The LFNGM's (K, M, q_0) and (L, N, r_0) over (Σ, Δ) are *equivalent* iff $M_T = N_T$.

Definition 3. A *Post correspondence system* over (Σ, Δ) is a pair of maps (g, h) from Σ into $\Delta^+ - \Delta$.[1] Maps g and h are extended to Σ^+ as follows. If $x, y \in \Sigma^+$,

$$g(xy) = g(x)g(y), \qquad h(xy) = h(x)h(y).$$

The associated problem is to tell whether there is a string $x \in \Sigma^+$ such that $g(x) = h(x)$.

Post [2] showed that no algorithm can solve the associated correspondence problem for all (Σ, Δ) and (g, h) over (Σ, Δ).

Given a correspondence system (g, h) over (Σ, Δ), with

$$n = \max(\,|\,g(a)\,|, |\,h(a)\,|: a \in \Sigma),$$

et G_n be the LFNGM $(\{p_0\}, J, p_0)$, where

$$J(p_0, a, p_0) = \{y \in \Delta^+: |\,y\,| \leq n\} \qquad (a \in \Sigma).$$

Then

$$J_T(x) = \{y \in \Delta^+: |\,x\,| \leq |\,y\,| \leq n\,|\,x\,|\} \qquad (x \in \Sigma^+).$$

We define LFNGM's $G_{ng} = (K, M, q_0)$ and $G_{nh} = (L, N, r_0)$ such that

$$M_T(x) = J_T(x) - \{g(x)\} \quad (x \in \Sigma^+), \qquad N_T(x) = J_T(x) - \{h(x)\} \quad (x \in \Sigma^+).$$

Recoding K and L, if necessary, so that they are disjoint, we pick $s_0 \notin K \cup L$ and set $G_{ngh} = (K \cup L \cup \{s_0\}, R, s_0)$ where, if $a \in \Sigma$,

$$R(t, a, t') = \begin{cases} M(q_0, a, t') & \text{if } t = s_0 \ \&\ t' \in K, \\ N(r_0, a, t') & \text{if } t = s_0 \ \&\ t' \in L, \\ M(t, a, t') & \text{if } t, t' \in K, \\ N(t, a, t') & \text{if } t, t' \in L, \\ \phi \text{ (empty set) otherwise.} \end{cases}$$

Then clearly

$$R_T(x) = M_T(x) \cup N_T(x) \qquad (x \in \Sigma^+);$$

i.e. if $x \in \Sigma^+$, then

$$R_T(x) \neq J_T(x) \quad \text{iff} \quad g(x) = h(x).$$

Or,

THEOREM. *The correspondence problem for (g, h) is solvable if the equivalence problem for G_n and G_{ngh} is solvable.*

[1] The usual formulation has g and h mapping Σ into Δ^+. Clearly there is no loss of generality in our formulation.

Construction of G_{ng}

Assuming Δ and $\{q_0, -, 0, +\}$ are disjoint, we set $K = \Delta \cup \{q_0, -, 0, +\}$. If $a \in \Sigma$, with $g(a) = b_1 \cdots b_u \quad (u \geq 2)$ and the b_i's in Δ, and if $c \in \Delta, \quad d \in \Delta - \{b_u\}$, and $q \in K$, then:

$$M(q_0, a, b_u) = \{b_1 \cdots b_{u-1}\},$$

$$M(q_0, a, d) = M(q_0, a, q_0) = \phi,$$

$$M(q_0, a, -) = \{y \in \Delta^+ : |y| < u\},$$

$$M(q_0, a, 0) = \{y \in \Delta^+ : |y| = u \;\&\; y \neq g(a)\},$$

$$M(q_0, a, +) = \{y \in \Delta^+ : u < |y| \leq n\},$$

$$M(c, a, q) = \{c\}M(q_0, a, q);$$

$$M(-, a, q) = \begin{cases} \Delta & \text{if} \quad q = -, \\ \phi & \text{otherwise;} \end{cases}$$

$$M(0, a, q) = \begin{cases} \{y \in \Delta^+ : |y| < u\} & \text{if} \quad q = -, \\ \{y \in \Delta^+ : |y| = u\} & \text{if} \quad q = 0, \\ \{y \in \Delta^+ : u < |y| \leq n\} & \text{if} \quad q = +, \\ \phi & \text{otherwise;} \end{cases}$$

$$M(+, a, q) = \begin{cases} \{y \in \Delta^+ : |y| = n\} & \text{if} \quad q = +, \\ \phi & \text{otherwise.} \end{cases}$$

LEMMA. *If $x \in \Sigma^+$, with $g(x) = c_1 \cdots c_v \quad (v \geq 2|x|)$, the c_i's in Δ, and $e \in \Delta - \{c_v\}$, then:*

$$M(q_0, x, c_v) = \{c_1 \cdots c_{v-1}\},$$

$$M(q_0, x, e) = M(q_0, x, q_0) = \phi,$$

$$M(q_0, x, -) = \{y \in \Delta^+ : |x| \leq |y| < v\},$$

$$M(q_0, x, 0) = \{y \in \Delta^+ : |y| = v \;\&\; y \neq g(x)\},$$

$$M(q_0, x, +) = \{y \in \Delta^+ : v < |y| \leq n|x|\}.$$

PROOF. Proof is by induction on $|x|$. For $|x| = 1$, the result follows directly from our definition of M.

Assuming that the result holds for $|x|$, we now show that it also holds for $|x| + 1$. Assume $a \in \Sigma$, with $g(a) = b_1 \cdots b_u \quad (u \geq 2)$, the b_i's in Δ, and $d \in \Delta - \{b_u\}$. Then $g(xa) = c_1 \cdots c_v b_1 \cdots b_u$, and

$$M(q_0, xa, b_u) = M(q_0, x, c_v)M(c_v, a, b_u) = \{c_1 \cdots c_v b_1 \cdots b_{u-1}\},$$

$$M(q_0, xa, d) = M(q_0, xa, q_0) = \phi,$$

$$M(q_0, xa, -) = M(q_0, x, c_v)M(c_v, a, -) \cup M(q_0, x, -)M(-, a, -)$$
$$\cup\, M(q_0, x, 0)M(0, a, -)$$

$$
\begin{aligned}
&= \{g(x)\}\{y \in \Delta^+: |y| < u\} \cup \{y \in \Delta^+: |x| \le |y| < v\}\Delta \\
&\qquad \cup \{y \in \Delta^+: |y| = v \ \&\ y \ne g(x)\}\{y \in \Delta^+: |y| < u\} \\
&= \{y \in \Delta^+: |xa| \le |y| < u + v\}, \\
M(q_0, xa, 0) &= M(q_0, x, c_v)M(c_v, a, 0) \cup M(q_0, x, 0)M(0, a, 0) \\
&= \{g(x)\}\{y \in \Delta^+: |y| = u \ \&\ y \ne g(a)\} \\
&\qquad \cup \{y \in \Delta^+: |y| = v \ \&\ y \ne g(x)\}\{y \in \Delta^+: |y| = u\} \\
&= \{y \in \Delta^+: |y| = u + v \ \&\ y \ne g(xa)\}, \\
M(q_0, xa, +) &= M(q_0, x, c_v)M(c_v, a, +) \cup M(q_0, x, 0)M(0, a, +) \\
&\qquad \cup M(q_0, x, +)M(+, a, +) \\
&= \{g(x)\}\{y \in \Delta^+: u < |y| \le n\} \\
&\qquad \cup \{y \in \Delta^+: |y| = v \ \&\ y \ne g(x)\}\{y \in \Delta^+: u < |y| \le n\} \\
&\qquad \cup \{y \in \Delta^+: v < |y| \le n|x|\}\{y \in \Delta^+: |y| = n\} \\
&= \{y \in \Delta^+: u + v < |y| \le n|xa|\}. \qquad \text{Q.E.D.}
\end{aligned}
$$

This lemma makes it clear that

$$M_T(x) = J_T(x) - \{g(x)\} \qquad (x \in \Sigma^+),$$

as was supposed to be the case. G_{nh} is constructed in similar fashion.

Remarks

An LFNGM (K, M, q_0) over (Σ, Δ) can be viewed as a simple kind of linear grammar [1]. We assume $c \notin \Sigma \cup \Delta$ and that K and $\Sigma \cup \Delta \cup \{c\}$ are disjoint. We write $yqx \to yzq'ax$, $yqx \to yzcax$ if $q, q' \in K$, $a \in \Sigma$, $x \in \Sigma^+ \cup \{\Lambda\}$, $y \in \Delta^+ \cup \{\Lambda\}$, and $z \in M(q, a, q')$. Letting $\Rightarrow$ be the transitive closure of $\to$, we say ycx is in the language generated by (K, M, q_0) iff $q_0 \Rightarrow ycx$, that is, iff $y \in M_T(x^\#)$.

This language has the interesting property that $|x| \le |y| \le n|x|$, where

$$n = \max(|z|: z \in M(q, a, q'), \text{ with } q, q' \in K \ \&\ a \in \Sigma).$$

Thus $\{z \in \Sigma^+: q_0 \Rightarrow ycz\}$ and $\{z \in \Delta^+: q_0 \Rightarrow zcx\}$ are both finite sets. This is the defining characteristic of c-finite languages.

The unsolvability of the LFNGM equivalence problem implies that there is no algorithm for determining whether two LFNGM languages are equal. Therefore,

COROLLARY. *There is no algorithm for determining whether two c-finite languages are equal.*

ACKNOWLEDGMENT. I am indebted to Seymour Ginsburg of the System Development Corporation for bringing problems (1) and (2) to my attention, and to George Sethares and Frank King of the AFCRL Data Sciences Laboratory for a careful reading of the manuscript and several useful suggestions.

REFERENCES

1. GINSBURG, S. *The Mathematical Theory of Context-Free Languages*. McGraw-Hill, New York, 1966.
2. POST, E. A variant of a recursively unsolvable problem. *Bull. Amer. Math. Soc. 52* (1946), 264–268.

RECEIVED SEPTEMBER, 1967; REVISED MARCH, 1968