THE GENERAL DECISION PROBLEM FOR MARKOV ALGORITHMS WITH AXIOM

C. E. HUGHES

Introduction* Let $\mathcal{M}_d$ denote the general decision problem for Markov algorithms with axiom. Of interest to us is whether or not this class of problems is as richly structured, with regard to degrees of unsolvability, as those classes studied in Hughes, Overbeek, and Singletary [2]. In this paper we shall present proofs which show this to be so. In particular we shall show that the general decision problem for the range of total recursive functions is many-one reducible to $\mathcal{M}_d$, and consequently that every r.e. many-one degree of unsolvability is represented by $\mathcal{M}_d$. Furthermore we shall show this result to be best possible, with regard to degree representation, in that every r.e. one-one degree is not represented by this family of decision problems. And finally we shall demonstrate a simple application of these results to the study of splinters.

Preliminaries A semi-Thue system $S$ is a pair $(\Sigma, P)$ where $\Sigma$ is a finite alphabet and $P$ is a finite set of rules each of which is of the form $\alpha \rightarrow \beta$, for $\alpha$ and $\beta$ words over $\Sigma$. For any arbitrary pair of words $W_1, W_2$ over $\Sigma$, we say that $W_2$ is an immediate successor of $W_1$ in $S$, denoted $(W_1, W_2)_S$, if there exist a pair of words $U$, $V$ over $\Sigma$ and a rule $\alpha \rightarrow \beta$ in $P$ such that $W_1 = U \alpha V$ and $W_2 = U \beta V$. $W_2$ is said to be derivable from $W_1$ in $S$, denoted $W_1 \vdash _S W_2$, if either

(i) $W_1 = W_2,$

or

(ii) there exists a finite sequence $V_1, \ldots, V_k$, where $k > 1$, of words over $\Sigma$ such that $W_1 = V_1$, $W_2 = V_k$, and $(V_i, V_{i+1})_S$, for $i = 1, \ldots, k - 1$.

A Markov algorithm $M$ is a pair $(\Sigma, P)$ where $\Sigma$ is a finite alphabet and $P = \{ \alpha_i R_i \beta_i \mid 1 \leq i \leq m \}$ is a finite ordered set of rules where

---

*This paper was written when the author was an NRC-NBS Postdoctoral Research Associate, 1971-72. He is now a member of the faculty at The Pennsylvania State University, University Park, Pennsylvania.

Received February 25, 1972
$m \geq 1$, $R_i \epsilon (\rightarrow, \rightarrow_{\cdot})$ and $\alpha_i$ and $\beta_i$ are words over $\Sigma$. A rule of the form $\alpha \rightarrow \beta$ is called a conclusive rule. Let $W_1$ and $W_2$ be arbitrary words over $\Sigma$. Then $W_2$ is the immediate successor of $W_1$ in $M$, denoted $(W_1, W_2)_M$, if there exists an $i$, $1 \leq i \leq m$, such that

(i) \hspace{1cm} W_1 = U\alpha_i V$ and $W_2 = U\beta_i V$ for some words $U$ and $V$ over $\Sigma$,

(ii) \hspace{1cm} there exists no pair of words $U', V'$ over $\Sigma$ such that the length of $U'$ is less than the length of $U$ and $W_1 = U'\alpha_i V'$,

(iii) \hspace{1cm} there exists no $j$, $1 \leq j < i$, such that $W_1 = U''\alpha_j V''$ for some words $U'', V''$ over $\Sigma$.

$W_2$ is said to be derivable from $W_1$ in $M$, denoted $W_1 \vdash_M W_2$, if either

(i) \hspace{1cm} $W_1 = W_2$,

or

(ii) \hspace{1cm} there exists a finite sequence $V_1, \ldots, V_k$, where $k > 1$, of words over $\Sigma$ such that $W_1 = V_1$, $W_2 = V_k$, $(V_i, V_{i+1})_M$, for $i = 1, \ldots, k - 1$, and no $(V_i, V_{i+1})_M$, for $1 \leq i \leq k - 2$, is the result of the application of a conclusive rule.

Let $G$ be a semi-Thue system or Markov algorithm and let $A$ be a fixed word over the alphabet of $G$. Then $G_A$ shall denote such a system with axiom. The decision problem for $G_A$ is the problem to decide, for an arbitrary word $W$ over the alphabet of $G$, whether or not $A \vdash_G W$ (written $\vdash_G W$ whenever $A$ is understood from context). The general decision problem for semi-Thue systems (Markov algorithms) with axiom is then the family of decision problems for all such systems.

Let $C_1$ and $C_2$ be two general decision problems. Then we say that $C_1$ is many-one (one-one) reducible to $C_2$ if there exists an effective mapping $\psi$ of the decision problems $p$ in $C_1$ into the decision problems $\psi(p)$ in $C_2$ such that $p$ and $\psi(p)$ are of the same many-one (one-one) degree of unsolvability. $C_2$ is said to represent every r.e. many-one (one-one) degree of unsolvability if the general decision problem for the range of total recursive functions, denoted $\mathcal{R}$, is many-one (one-one) reducible to $C_2$.

**Background Results** In the next section we have need of the following theorem concerning semi-Thue systems with axiom.

**Theorem 1** There exists an effective procedure $\psi_1$ which, when applied to an arbitrary total recursive function $f$, produces a semi-Thue system $S$ and a word $A$ over the alphabet of $S$ such that

(i) \hspace{1cm} the decision problem for the range of $f$ is of the same many-one degree as that for $S_A$;

(ii) \hspace{1cm} there is no non-trivial derivation of $A$ from $A$. That is, there exists no word $W$ over the alphabet of $S$ such that $(A, W)_S$ and $W \vdash_S A$;

(iii) \hspace{1cm} if $\vdash_S W$ by a non-trivial derivation and there exist $W', W''$ such that $(W', W)_S$ and $(W'', W)_S$ then $W' = W''$ and $\vdash_S W'$. Stated differently this says
that the semi-Thue system $S^{-1}_S$, whose alphabet is that of $S$ and which contains the rule $\beta \rightarrow a$ if and only if $a \rightarrow \beta$ is a rule of $S$, is deterministic\(^1\) over words which are non-trivially derivable from $A$ in $S$;

(iv) if $\downarrow W$ and $(W, W')_S$ by some rule of $S$ and $(W, W')_S$ by the same rule then $W' = W$;

(v) the word of length zero is not derivable from $A$ in $S$. In particular this means that $A$ may not be the empty word;

(vi) no rule $a \rightarrow \beta$ of $S$ is such that either $a$ or $\beta$ is the word of length zero.

Proof: In [5] Overbeek showed that $R$ is many-one reducible to the general halting problem for Turing machines. Following this, Hughes and Singletary [3], Lemma 3, demonstrated an effective procedure which, when applied to an arbitrary Turing machine $T$, produces a semi-Thue system with axiom (denoted $S_{halt}$ in their paper), which system satisfies properties (ii) through (vi) above and whose decision problem is of the same many-one degree as the halting problem for $T$. These two results may then be combined to provide a proof of the desired theorem. Q.E.D.

Reduction of $R$ to $M_b$ In this section we shall demonstrate a uniform effective procedure $\psi_2$ which, when applied to an arbitrary semi-Thue system $S$ with axiom $A$ that satisfies properties (ii) through (vi) of Theorem 1, produces a Markov algorithm $M$ with axiom $B$ such that the decision problem for $S_A$ is of the same many-one degree as that for $M_B$.

Let $S_A$ be a semi-Thue system with axiom which satisfies properties (ii) through (vi) of Theorem 1. Further, let the alphabet of $S$ be $\Sigma = \{\#, 1, \ldots, n\}$ and let the rule set of $S$ be $P = \{a_i \rightarrow \beta_i \mid 1 \leq i \leq \rho\}$. We define the Markov algorithm $M = (\Sigma', P')$ as follows:

$$\Sigma' = \Sigma \cup \{1, \ast\} \cup \{R_i, l_i, L_i, f_i, g_i, h_i \mid 1 \leq i \leq \rho\} \cup \{\$\, Q, r, e_1, e_2\};$$

$P'$ consists of the rules defined below, where a set of rules labelled (i) may have any internal order provided these rules follow all those in sets labelled (j), where $j < i$, and precede all rules in sets labelled (j), where $j > i$.

1. A system $S$ is deterministic over a word $W$ if and only if there exists at most one $W'$ such that $(W, W')_S$. 

\[\begin{align*}
(1) & \; \$ \rightarrow Q \\
(2) & \; R_i a_i \rightarrow \beta_i r \\
(3) & \; R_i a_j \rightarrow a_j R_i \\
(4) & \; R_i \ast \rightarrow l_i \ast \\
(5) & \; r a_i \rightarrow a_j r \\
(6) & \; r \ast 11 \rightarrow e_2 \ast 1 \\
(7) & \; r \ast 1 \rightarrow e_i \ast \\
(8) & \; a_i e_j \rightarrow e_j a_i \\
(9) & \; e_i \rightarrow R_j \\
(10) & \; a_i e_1 \rightarrow e_2 a_i
\end{align*}\]
We now wish to show that the decision problem for $S_A$ is of the same many-one degree as that for $M$ with axiom $A^*$. Before doing this we shall present the algorithm of which $M$ is an implementation. This, we believe, will help to make the subsequent proofs more understandable.

The Basic Algorithm

(1) Let the word we are currently working on be of the form $¥PF^*$ for $W$ a word over $\Sigma$. Then $W$ is a word derivable from $A$ in $S$. Generate $QW^*$.

(II) Let the word we are currently working on be of the form $R_i^W^*m$ for $1 \leq i \leq \rho$, $m \geq 1$, $1^m$ being a shorthand notation for the sequence of $m$ 1's, and $W$ a word over $\Sigma$.

case a) If the $i$'th rule of $S$ applies to $W$ and $m > 1$ then generate $R_i^W^*1^{m-1}$, where $(W, W')_S$ by the $i$'th rule.

case b) If the $i$'th rule of $S$ applies to $W$ and $m = 1$ then generate $¥W^*$, where $(W, W')_S$ by the $i$'th rule.

case c) If the $i$'th rule of $S$ does not apply to $W$ and $i < \rho$ then generate $R_{i+1}^W1^m$.

case d) If the $i$'th rule of $S$ does not apply to $W$ and $i = \rho$ then generate $QW^*1^m$.

(III) Let the word we are currently working on be of the form $QW^*1^m$ for $m \geq 0$, $1^m$ denoting a sequence of $m$ 1's, and $W$ a word over $\Sigma$.

case a) If $W \equiv A$ then generate $R_1A^*1^{m+1}$.

case b) If $W \neq A$ and $(W', W)_S$ by some rule $i$, $i < \rho$, then generate $R_{i+1}^W1^{m+1}$.

case c) If $W \neq A$ and $(W', W)_S$ by rule $\rho$, then generate $QW^*1^{m+1}$.

case d) If $W \neq A$ and it is not the case that there exists a $W'$ such that $(W', W)_S$, then stop.

The intention of the above algorithm, when started on the word $¥A^*$, is
to generate words of the form $W^*$, when and only when $W$ is derivable from $A$ in $S$. Essentially if we think of (1) as performing the additional task of outputting $W$ whenever it is entered with a word of the form $W^*$ then this algorithm would simply enumerate the set of words derivable from $A$ in $S$. To see this, observe that if we input $A^*$ to this procedure then we will immediately derive $QA^*$ and then $R_1A^*1$. From this we will, by successive applications of II, I, and III, generate $W^*$ for every word $W$ such that $(A, W)_S$. After generating every immediate successor of $A$, in the order determined by our previous ordering of the rules of $S$, the algorithm will start working over the word $R_1A^*11$. In general, the basic algorithm will, when started over the word $R_1A^*1^m$, generate $W^*$ for every word $W$ such that $\vdash_3 W$ by a derivation of length $m$. After generating all such words it will start working over $R_1A^*1^m1$.

The reader should now be convinced that the basic algorithm is one which, in essence, "flattens" the rooted graph induced by $S_A$. The special properties of $S_A$ are essential to this for they ensure that this rooted graph is a rooted tree and therefore devoid of nodes with indegree greater than one. We may also note that this algorithm never terminates when started over $A^*$. This may be seen by observing that the only way halting may occur is by the application of III.d. But this would only arise if there existed some word $W$, $W \neq A$, such that $\vdash_3 W$ and there is no $W'$ such that $(W', W)_S$. Clearly this is impossible.

We shall now proceed with our proof that the decision problem for $S_A$ is of the same many-one degree as that for $M_{A^*}$.

Lemma 1 The decision problem for $S_A$ is many-one reducible to that for $M_{A^*}$.

Proof: This may be seen to be true if we can verify that, for an arbitrary $W$ over the alphabet of $S$, $\vdash_3 W$ if and only if $\vdash_M W^*$. But this follows immediately since $M$ implements our basic algorithm where the rules of $M$ may be corresponded to the steps of the basic algorithm as follows:

<table>
<thead>
<tr>
<th>rules</th>
<th>step</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1}</td>
<td>I</td>
</tr>
<tr>
<td>{2, 3, 5, 6, 8, 9}</td>
<td>II.a</td>
</tr>
<tr>
<td>{2, 3, 5, 7, 10, 11}</td>
<td>II.b</td>
</tr>
<tr>
<td>{3, 4, 12, 13}</td>
<td>II.c</td>
</tr>
<tr>
<td>{3, 4, 12, 14}</td>
<td>II.d</td>
</tr>
<tr>
<td>{15}</td>
<td>III.a</td>
</tr>
<tr>
<td>{16, 17, 18, 19, 20, 21, 22, 23, 25, 26}</td>
<td>III.b</td>
</tr>
<tr>
<td>{16, 17, 18, 19, 20, 21, 22, 24, 25, 26}</td>
<td>III.c</td>
</tr>
<tr>
<td>{16, 18, 19, 25, 26}</td>
<td>III.d</td>
</tr>
</tbody>
</table>

Q.E.D.

Lemma 2 The decision problem for $M_{A^*}$ is many-one reducible to that for $S_A$.

Proof: Let $W$ be an arbitrary word over $\Sigma'$. It should be clear that we may
assume, without loss of generality, that \( W \) is of the form \( W'\ast 1^m \) for \( m \geq 0 \) and \( W' \) a word over \( (\Sigma' - \{\ast, 1\}) \) which contains exactly one occurrence of a letter from \( ((\Sigma' - \Sigma) - \{\ast, 1\}) \). For if this were not so then \( \uparrow_M W \) (that is, \( W \) would not be derivable from \( \$A\ast \) in \( M \)). Hence we shall hereafter assume \( W \) to be of this form. Now we may, with the aid of an oracle for deciding the decision problem for \( S_A \), determine whether or not \( \uparrow_M W \) by the following case analysis:

(a) Assume \( W \) contains an occurrence of the letter \$. If \( W \) is not of the form \( Y\ast \) for \( Y \) a word over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then \( \uparrow_M W \) if and only if \( \uparrow_Y W \).

(b) Assume \( W \) contains an occurrence of the letter \( R_i \) for \( 1 \leq i \leq \rho \). If \( W \) is not of the form \( YR_i Z\ast 1^m \) for \( m \geq 1 \) and \( Y \) and \( Z \) words over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then check to see if \( R_i YZ\ast 1^m \) \( \uparrow_M W \) in exactly the length of \( Y \) steps. If it does not then \( \uparrow_M W \). If it does then \( \uparrow_M W \) if and only if \( \uparrow_Y YZ \).

(c) Assume \( W \) contains an occurrence of \( Y_i \) for \( 1 \leq i \leq \rho \). If \( W \) is not of the form \( YR_i Z\ast 1^m \) for \( m \geq 1 \) and \( Y \) and \( Z \) words over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then check to see if \( R_i YZ\ast 1^m \) \( \uparrow_M W \) in exactly the length of \( Y \) plus twice the length of \( Z \) plus one steps. If it does not then \( \uparrow_M W \). If it does then \( \uparrow_M W \) if and only if \( \uparrow_Y Y'Z' \).

(d) Assume \( W \) contains an occurrence of the letter \( r \). If \( W \) is not of the form \( YrZ\ast 1^m \) for \( m \geq 0 \) and \( Y \) and \( Z \) words over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then check to see if there exists an \( i \), \( 1 \leq i \leq \rho \), and a pair of words \( Y' \), \( Z' \) over \( \Sigma \) such that \( Y = Y'\beta_i Z' \). If there does not exist exactly one triple \( i \), \( Y' \), \( Z' \) satisfying the above requirement then \( \uparrow_M W \), else \( \uparrow_W W \) if and only if \( \uparrow_Y Y'Z' \).

(e) Assume \( W \) contains an occurrence of the letter \( e \). If \( W \) is not of the form \( Ye\ast 1^m \) for \( m \geq 1 \) and \( Y \) a word over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then \( \uparrow_M W \) if and only if \( \uparrow_Y YZr\ast 1^m+1 \) and our analysis returns to case (d).

(f) Assume \( W \) contains an occurrence of the letter \( e_i \). If \( W \) is not of the form \( Ye_i Z\ast 1^m \) for \( m \geq 1 \) and \( Y \) and \( Z \) words over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then \( \uparrow_M W \) if and only if \( \uparrow_M YZr\ast 1 \) and our analysis returns to case (d).

(g) Assume \( W \) contains an occurrence of the letter \( Q \). If \( W \) is not of the form \( QY\ast 1^m \) for \( m \geq 0 \) and \( Y \) a word over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then \( \uparrow_M W \) if and only if \( \uparrow_Y Y \).

(h) Assume \( W \) contains an occurrence of the letter \( L_i \) for \( 1 \leq i \leq \rho \). If \( W \) is not of the form \( YL_i Z\ast 1^m \) for \( m \geq 0 \) and \( Y \) and \( Z \) words over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then \( \uparrow_M W \) if and only if \( \uparrow_M YZr\ast 1 \) and our analysis returns to case (d).

(i) Assume \( W \) contains an occurrence of the letter \( g_i \) for \( 1 \leq i \leq \rho \). If \( W \) is not of the form \( Yg_i Z\ast 1^m \) for \( m \geq 0 \) and \( Y \) and \( Z \) words over \( \Sigma \) then \( \uparrow_M W \). If \( W \) is of this form then \( \uparrow_M W \) if and only if \( \uparrow_M YZr\ast 1^m \) and our analysis returns to case (h).

(j) Assume \( W \) contains an occurrence of the letter \( g_p \) then \( \uparrow_M W \).
(k) Assume $W$ contains an occurrence of the letter $f_i (1 \leq i \leq \rho)$. If $W$ is not of the form $Yf_iZ\ast 1^n$ for $m \geq 0$ and $Y$ and $Z$ words over $\Sigma$ then $\mathcal{V}_M W$. If $W$ is of this form then check to see if there exists a pair of word $Y', Z'$ over $\Sigma$ such that $Y \equiv Y' a_i Z'$. If there does not exist exactly one pair $Y', Z'$ satisfying the above requirement then $\mathcal{V}_M W$, else $\mathcal{V}_M W$ if and only if $\mathcal{V}_M L_i Y' \beta_i Z' \ast 1^m$ and our analysis returns to case (h).

(l) Assume $W$ contains an occurrence of the letter $h_i (1 \leq i \leq \rho)$. If $W$ is not of the form $Yh_iZ\ast 1^m$ for $m \geq 1$ and $Y$ and $Z$ words over $\Sigma$ then $\mathcal{V}_M W$. If $W$ is of this form then $\mathcal{V}_M W$ if and only if $\mathcal{V}_M YZf_i \ast 1^{m-1}$ and our analysis returns to case (k).

The proof may now be seen to be complete by simply observing that, for any given word $W$, the above decision procedure asks at most one question of the oracle for $S_A$ and, if it chooses to ask a question, reports the oracle's answer faithfully. Q.E.D.

Theorem 2 Each of the following holds

(I) $\mathcal{R}$ is many-one reducible to $\mathcal{M}_A$,

(II) Every r.e. many-one degree is represented by $\mathcal{M}_A$.

Proof: (I) is an immediate consequence of Lemmas 1 and 2 and Theorem 1. (II) follows from (I) by the definition of what it means for a general decision problem to represent every r.e. many-one degree. Q.E.D.

Corollary 1 Define a restricted Markov algorithm to be one which does not contain any conclusive rules. Then Theorem 2 holds with $\mathcal{M}_A$ replaced by $\mathcal{M}'_A$ where $\mathcal{M}'_A$ is the general decision problem for restricted Markov algorithms with axiom.

Proof: Immediate from the fact that the Markov algorithms defined by our procedure $\psi_2$ are always restricted. Q.E.D.

The One-One Degrees of $\mathcal{M}_A$ We shall now prove that our result for $\mathcal{M}_A$ is best possible with regard to degree representation. In order to do this we shall prove that no instance of $\mathcal{M}_A$ is of the same one-one degree as a simple set [6], p. 298. This theorem and its proof are a particularization of a more general theorem of W. E. Singletary. Other papers which have employed his result include [1], [2], and [3].

Theorem 3 It is not the case that every r.e. one-one degree is represented by $\mathcal{M}_A$.

Proof: Let $M_A$ be a Markov algorithm with axiom whose decision problem is recursively unsolvable. Then we may deduce the existence of some word $W_o$ over the alphabet of $M$ such that $\mathcal{V}_M W_o$, $D = \{ W \mid W_o \mathcal{V}_M W \}$ is infinite and no

---

2. By reporting the answer faithfully we mean that the procedure may not continue to compute after asking a question of the oracle nor may it apply any Boolean function to the oracle's answer.
THE GENERAL DECISION PROBLEM 215

word contained in \( D \) is derivable from \( A \). For assume no such \( W_0 \) exists then the decision problem for \( M_A \) may be decided as follows: Let \( W \) be an arbitrary word over the alphabet of \( M \). Build two list of words in the following manner: Stage 0, put \( W \) in list 1 and \( A \) in list 2. Stage \( n + 1 \), put the word derivable from \( W \) in \( n + 1 \) steps into list 1, if any such word exists. Do the same for list 2 with respect to \( A \). Continue this process until either (1) list 1 contains all words derivable from \( W \) in \( M \); or (2) list 1 and list 2 contain some word in common. By our assumption one of these cases must arise. Now if (1) occurs then \( \nu_M W \) since \( A \) must have an infinite number of descendants in order for the decision problem for \( M_A \) to be unsolvable. If (2) first occurs due to \( W \) being placed in list 2 then \( \sim_M W \). Otherwise (2) first occurs because there exists some \( W' \) such that \( W \sim_M W' \), non-trivially, and \( A \sim_M W' \) in a derivation in which \( W \) does not arise (note: this includes the possibility that \( W \sim_M A \), non-trivially). But then \( \nu_M W \). For, if \( \sim_M W \), then \( M \) would loop when started on \( A \) and hence \( M_A \) would be solvable. Thus any r.e. set of the same one-one degree as the decision problem for \( M_A \) must be non-simple and hence every r.e. one-one degree is not represented by \( M_A \). Q.E.D.

Degrees of Splinters Let \( f \) be a total recursive function and let \( a_0 \) be a natural number. Then \( \{ x \mid \exists n[f^n(a_0) = x] \} \), where \( f^0(a_0) = a_0 \) and \( f^{n+1}(a_0) = f(f^n(a_0)) \), is called a splinter. Splinters were first defined by Ullian [7] and were subsequently studied by Myhill [4]. One of the results of their research was the proof that every r.e. many-one degree is represented by the general decision problem for splinters. We shall now show that our results for \( M_A \) provide us with an independent proof of this.

Theorem 4 Every r.e. many-one degree is represented by the general decision problem for splinters.

Proof: Let \( M \) be an arbitrary restricted Markov algorithm, let \( A \) be a word over the alphabet of \( M \), and let \( g_M \) be a G\ödel numbering of the words of \( M \) onto the natural numbers. Define the total recursive function \( f_M \) as follows:

\[
f_M(x) = \begin{cases} 
x & \text{if } g_M^{-1}(x) \text{ has no immediate successor} \\
y & \text{if } g_M^{-1}(y) \text{ is the immediate successor of } \\
& g_M^{-1}(x) \text{ in } M. 
\end{cases}
\]

Let \( a_0 = g_M(A) \). Then clearly \( \{ x | \exists n[f^n_M(a_0) = x] \} \) is just the set of G\ödel numbers of all words derivable from \( A \) in \( M \). Hence the decision problem for \( M_A \) is of the same many-one (in fact one-one) degree as that for the splinter arising from \( f_M \) and \( a_0 \). But then, since \( M \) and \( A \) were chosen arbitrarily, we have that \( M_A \) is many-one reducible to the general decision problem for splinters and this theorem is proved in light of Theorem 2, Corollary 1. Q.E.D.
REFERENCES


*Institute for Computer Sciences and Technology*

*National Bureau of Standards*

*Washington, D.C.*