COMPLETE LANGUAGE TABLES

U. Brandt and H.K.-G. Walter

Keywords: Languages, language tables, defect, complete tables, domino problems, decidability.

Abstract: A language table is a two dimensional data structure, normally a square, which is built up like a "crossword puzzle" associated to a language L. The paper deals with the problem to construct to a given language L language tables where no zero entries (i.e., entries different from letters) occur, so called complete tables. We show for example that whether or not there exists a language table for L of size n for some n, is undecidable for regular languages even defined over a two letter alphabet, though it is decidable for standard events. The proof shows that language tables are more powerful than dominoes because we can encode tilings of the plane, squares, etc., by complete language tables of very simple languages.

1. INTRODUCTION

Language tables are two dimensional data structures which are built in "crossword puzzle" manner. That means, given a language L over some alphabet X, we consider connected squares of empty entries or letter entries such that reading rows from left to right or columns from top to bottom we meet either words of L or isolated letters. Language tables therefore define patterns constituted of various components such that horizontally and vertically the structure is controlled by the syntactical structure of the given language.

There are various natural questions associated with such a device. One measure is to count the number of empty places, the defect of a table. This measure defines in some sense the compactness of possible patterns. The most compact pattern is a table without any empty entries, a so called complete table. In this paper we study decidability problems on completeness; in [2] the combinatorical aspects of this measure is investigated.

We focus our interst to the following decidability problems.

(1) The n-CTP (complete table problem): Given n, is there a complete table for L of size n?

(2) The CTP: Is there an n, such that there exists a complete L-table of size $n > 1$?

(3) The infinite CTP: Is there an infinite sequence of complete L-tables increasing in size?

If X is a one-letter alphabet, all three problems can be rewritten easily to very well-known decidability problems. The situation is completely different for greater alphabets. Our main result is that the CTP and the infinite CTP are undecidable for regular sets, even defined over two letter alphabets, though decidable for standard events, whereas the n-CTP is NP-complete for regular sets and contextfree languages.

The method to prove these results relies heavily on the fact that "tiling" problems give rise to construct complete tables for very simple (i.e., regular) languages. We use the paper of R.M. Robinson [3], where Turing machine computations are encoded by tilings of the plane, squares, etc.

2. BASIC NOTATIONS AND PRELIMINARY CONSIDERATIONS

We assume that the reader is familiar with the basic definitions and results of formal language theory (s. for ex. [1]).

Consider an alphabet X and a symbol $0 \notin X$, which will represent empty entries. We are dealing with (n,n)-matrices A over $X \cup 0$.

If A is such a matrix, then

$$(RA)_i = A[i,1]\ldots A[i,n] \text{ and}$$
$$(CA)_i = A[1,i]\ldots A[n,i] \text{ for } 1 \le i \le n.$$

Obviously, $(CA)_i = (RA^T)_i$ if A^T is the transposed matrix. To any such matrix we associate the graph G_A with vertices

$$\{(i,j)/1 \le i,j \le n\}$$

and edges

$$\{((i,j),(k,\ell))/|i-k|+|j-\ell|=1, A[i,j]\neq 0 \text{ and } A[k,\ell]\neq 0\}.$$

Now, let $L \subseteq X^*$ be a language. We assume $X \subseteq L$.

Definition:

(1) A is an R-L-table (of size n) if and only if G_A is connected and

$$(RA)_i \in 0^*\cdot(L\cdot 0^+)^*\cdot 0^* \text{ for } 1 \le i \le n.$$

(2) A is a C-L-table (of size n) if and only if A^T is an R-L-table.

(3) A is an L-table (of size n) if and only if A is both an R-L-table and a C-L-table.

Example: Consider the Dyck-language $D_1 \subseteq \{(,)\}^*$. Let $L = D_1 \cup \{(,)\}$. Then

$$A_1 = \begin{bmatrix} 0 & 0 & (& 0 & (&) \\ 0 & (&) & (&) & 0 \\ (&) & (&) & (&) \\ 0 & (&) & (&) & 0 \\ (&) & (&) & (&) \\ ) & 0 &) & 0 &) & 0 \end{bmatrix} \quad \text{and} \quad A_2 = \begin{bmatrix} (& (&) & (&) &) \\ (&) & 0 & (& 0 & 0 \\ ) & 0 & 0 &) & 0 & (\\ ) & 0 & (&) & (&) \\ (&) & 0 & (&) & 0 \\ ) & 0 & (&) & 0 & 0 \end{bmatrix}$$

are both L-tables.

We are interested in filling tables in the best possible way. Therefore we define for $w \in (X \cup \{0\})^*$,

$|w|_0$ = number of occurences of 0 in w,

$|w|$ = length of w

and

$$\mathrm{def}(A) = \Sigma_{i=1}^{n} |(RA)_i|_0 .$$

We call A complete if $\mathrm{def}(A) = 0$.

Let us start considering the case $X = \{a\}$. If there is a word w in $L \subseteq \{a\}^*$ then, obviously, there is a complete L-table of size $|w|$ ($A[i,j] = a$ for $1 \le i,j \le$ $\le |w|$).

Hence we can conclude the following facts:

(1) To a given n there exists a complete L-table A of size n if and only if there is a word $w \in L$ with $|w| = n$.

(2) There exists a complete L-table A of size $n > 1$ if and only if $L \cap \{w/|w| = n\} \neq \emptyset$.

(3) There exists an infinite sequence $A_1, A_2, \ldots$ of complete L-tables A_i of size n_i with $n_1 < n_2 < \ldots$ if and only if L is infinite.

Thus our three decidability problems are completely reduced to the very well-known decidability problems for languages $L \subseteq \{a\}^*$.

Now consider $L \subseteq X^*$ for an arbitrary X. Then the n-CTP is decidable for L if the question "$w \in L$?" is decidable for every $w \in X^*$ with $|w| = n$. For we can determine in this case

$$L_n = L \cap \{w/|w| = n\}$$

and check whether or not a complete L_n-table of size n exists. Obviously, the n-CTP for L is in NP if there is a recognition algorithm for L of polynomial time complexity.

Before we start to prove our main results, we show that the CTP and the infinite CTP are decidable for standard events.

LEMMA 1. *If* $L = S_{\alpha\gamma\omega}$ *is a standard event, then there is a complete* L*-table of size* n+1 *if and only if there are two words* w,v *and a letter* x *with* $wx \in \alpha^+ \cap L$, $xv \in \omega^+ \cap L$ *and* $|w| = |v| = n$.

P r o o f. Suppose there is a complete L-table A of size n+1. Consider $(RA)_1 = w \cdot A[1,n+1]$ and $(CA)_{n+1} = A[1,n+1] \cdot v$. Using $x = A[1,n+1]$ we get immediately the result.

On the other hand, let wxv be given according to the assumption. Let $x = x_1 \dots x_n$, $v = y_1 \dots y_n$, then build the following table

$$\begin{bmatrix} x_1 & x_2 \ldots x_n & x \\ x_2 & x_3 \ldots x & y_1 \\ \vdots & & \vdots \\ x_n & x \ \ldots y_{n-2} & y_{n-1} \\ x & y_1 \ldots y_{n-1} & y_n \end{bmatrix}$$

which is obviously a complete L-table.

Since it is decidable, whether or not this kind of word wxv exists in L for every standard event L, we get the following

COROLLARY. *The CTP is decidable for standard events.*

In the following denote by k_L the constant of the pumping lemma for the regular set L (for example, derived from the minimal acceptor).

LEMMA 2. *If* $L = S_{\alpha\gamma\omega}$ *is a standard event then the following two statements are equivalent:*

(i) *There is a sequence* $(A_i)_{i=0}^{\infty}$ *of complete L-tables of increasing size* n_i.

(ii) *There exist two words* w,v *and a letter* x *with* $wx \in \alpha^+ \cap L, xv \in \omega^+ \cap L$, $|wx| > k_L$ *and* $|xv| > k_L$.

P r o o f. (i) => (ii): By Lemma 1 we can assign to every i, words w_i, v_i and a letter x_i such that $w_i x_i \in \alpha^+ \cap$ $\cap L$ and $x_i v_i \in \omega^+ \cap L$ and $|w_i x_i| = |x_i v_i| = n_i$. Thus

(ii) follows immediately choosing $n_i > k_L$.

(ii) $\Rightarrow$ (i): Let w,v,x be given according to the assumption. Then, by the pumping lemma for regular sets:

(1) $wx = w_1w_2w_3, |w_2| > 0$ and $\forall\, n \geq 1$: $w_1w_2^nw_3 \in L$,

(2) $xv = v_1v_2v_3, |v_2| > 0$ and $\forall\, n \geq 1$: $v_1v_2^nv_3 \in L$.

Consider $i \in N$. Then there exist $j,\ell \geq 1$ with $|w_1w_2^jw_3| \geq i$ and

$$|w_1w_2^jw_3| \leq |v_1v_2^\ell v_3|.$$

We get $v_1v_2^\ell v_3 = xv'$ for some appropriate v' and $xv' \in \omega^+ \cap L$ because

$$xv = v_1v_2v_3 \in \omega^+ \cap L .$$

Furthermore, $xv'' \in \omega^+ \cap L$ for every v'' with $v''u = v'$ for some u. Choose v'' such that $|xv''| = |w_1w_2^jw_3|$. Analogously, $w_1w_2^jw_3 = w'x$ for some w' and $w'x \in \alpha^+ \cap L$.

Now, replacing in the construction of Lemma 1 the words w and v by the word w' and v'', we get a complete L-table of size $n_i \geq i$. Since there exists for every i, a complete L-table of size $n_i \geq i$, statement (i) follows immediately.

Since it is decidable whether or not L fulfills assumption (ii) for every standard event L, we get the following

COROLLARY. *The infinite CTP is decidable for standard events.*

3. UNDECIDABILITY RESULTS

We want to show that the CTP and the infinite CTP are both undecidable for regular sets. To do this, we connect both problems with domino problems of tiling the plane respectively squares of arbitrary size. Following R.M. Robinson [3], we can encode Turing machine computations by "tiling" the plane with an associated set of dominoes. We don't need the construction of this domino set but the reader can visualize our method considering the appropriate tiling problems. We sharpen the general undecidability result for regular sets over $X = \{a,b\}$.

Our proof includes three parts:

I. Instead of using a single language L for both, rows and columns, we use a pair of languages (L_1,L_2) controlling separately columns and rows.

II. We encode Turing machine computations by tables, where rows and columns are controlled by two languages L_1 and L_2 which are essentially standard events. Here the method of R.M. Robinson is used.

III. We encode complete tables over arbitrary alphabets to complete tables over the two letter alphabet $\{a,b\}$.

Let us start with the first step. We extend the definition of an L-table to pairs (L_1, L_2) of languages: an (n,n)-matrix A (over $X \cup O$) is an (L_1, L_2)-table if and only if one of the following two conditions holds:

(i) A is an R-L_1-table and a C-L_2-table, or

(ii) A is an R-L_2-table and a C-L_1-table.

Now, we show that we can merge L_1 and L_2 together to one language L preserving regularity and completeness. This is done by an endmarker technics. Consider a set of letters

$$H = \{\#_1, \#_2, \#_3, \#_4, \$, ¢\}$$

with $H \cap X = \emptyset$. Let $X' = X \cup H$.

Define L by

$$L = \#_1\$^* \#_2 \cup \#_3 \$^* \#_4 \cup \$L_1\$ \cup$$
$$\#_1 ¢^* \#_3 \cup \#_2 ¢^* \#_4 \cup ¢ L_2 ¢ .$$

Observe that L is regular if and only if L_1 and L_2 are both regular. Obviously, any complete (L_1, L_2)-table of size n transfers to a complete L-table of size n+2.

Now, we show the converse correspondence, namely, that every complete L-table of size $n > 2$ can be transformed into a complete (L_1, L_2)-table. Consider a complete L-table A of size $n > 2$. All words in L must start with $\#_1, \#_2, \#_3, \$$ or ¢. Assume $A[1,1] \in \{\$, ¢\}$. Then $A[1,2]=0$

or $A[1,2] \in X$ and therefore $A[2,2] = 0$.

Now, assume $A[1,1] \in \{\#_2,\#_3\}$. Then $A[1,j] = \#_4$ for some $1 < j \leq n$. Hence $A[2,j] = 0$. In conclusion $A[1,1] = \#_1$ because $def(A) = 0$. Thus

$$(RA)_1 \in \#_1\$^* \#_2 \cup \#_1 \not\subset^* \#_3 .$$

First case: Let $(RA)_1 \in \#_1\$^* \#_2$. Then

$$(CA)_1 \in \#_1\$^* \#_2 \cup \#_1 \not\subset^* \#_3 .$$

Assume $(CA)_1 \in \#_1S^* \#_2$. Then we get $(RA)_n \in \#_2 \not\subset^* \#_4$, i.e., $(CA)_2 = \$w \not\subset$ for some $w \in X' \cup 0$. Since there is no word in L starting with \$ and terminating with $\not\subset$, $A[j,2] = 0$ for some $1 < j < n$, and therefore $def(A) > 0$ - a contradiction. Thus we get $(CA)_1 \in \#_1 \not\subset^* \#_3$, i.e., $(RA)_n \in$ $\in \#_3\$^* \#_4$ and $(CA)_n \in \#_2 \not\subset^* \#_4$. Since $def(A) = 0$, $(CA)_\ell \in \$L_1\$$ and $(RA)_\ell \in \not\subset L_2 \not\subset$ for $1 < \ell < n$. Hence by deleting the borderlines we obtain an (L_1,L_2)-table A' of size n-2 with $def(A') = 0$.

The *second case,* $(RA)_1 \in \#_1 \not\subset^* \#_3$, is symmetric.

Thus we have proven

LEMMA 1. *To any pair of languages* $L_1,L_2 \subseteq X^*$ *we can construct a language* L *such that the following statements are equivalent for any* $n > 2$:

(1) *There exists a complete* (L_1,L_2)*-table of size* n.

(2) *There exists a complete* L*-table of size* n-2.

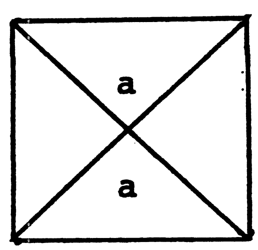
Furthermore, L *is regular if and only if* L_1 *and* L_2 *are both regular.*

The next step in our proof is the encoding of Turing machine computations by language tables.

We follow R.M. Robinson. Consider a one tape Turing machine T which moves in every step and stops with an accepting state, otherwise it doesn't stop (s. [3]). First we describe the alphabet and then the two languages L_1, L_2, which control the desired table in such a way that a computation is encoded.

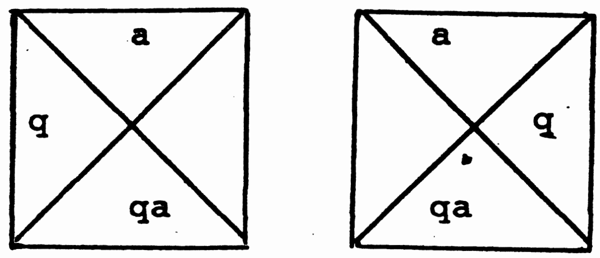
Let B_T denote the alphabet, Q_T the set of states and Δ_T the (deterministic) program of T. Furthermore we indicate the initial state by in(T), the "accepting" state by stop(T) and the empty-cell symbol by $\emptyset$. We assume in(T) $\neq$ stop(T). Now, the alphabet of L_1 and L_2 consists of three parts L, M, A representing

L: "letter tiles"



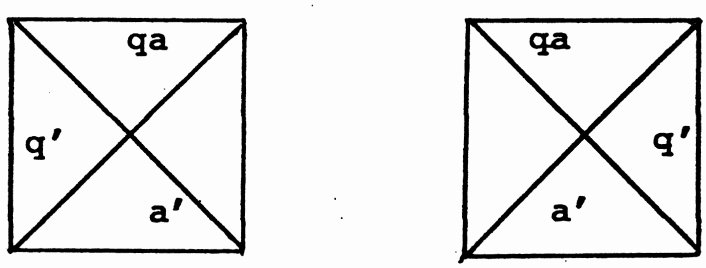
$a \in B_T$

M: "merging tiles"



$q \in Q_T, \quad a \in B_T$

A: "action tiles"



$qaa'q'L \in \Delta_T$ $\qquad$ $qaa'q'R \in \Delta_T$

Let the functions left, right, top and bottom from $\mathcal{L} \cup \mathcal{M} \cup \mathcal{A}$ into $(Q_T \cup B_T)^*$ determine the word on the left, right, top and bottom of every tile.

The design of the language pair encoding computations can be derived by the following picture

← tape →

	0	*L*	*L*	*L*	*M*	*A*	*L*	*L*
	1	*L*	*L*	*L*	*A*	*M*	*L*	*L*
	2	*L*	*L*	*L*	*L*	*A*	*M*	*L*
time ↓	3	*L*	*L*	*L*	*L*	*M*	*A*	*L*
	4	*L*	*L*	*L*	*M*	*A*	*L*	*L*
	5	*L*	*L*	*M*	*A*	*L*	*L*	*L*
	6	*L*	*M*	*A*	*L*	*L*	*L*	*L*

By this picture the "tape" language L_1 is given by $L_1 = \mathcal{L}^* \cdot \{xy/((x \in \mathcal{M} \wedge y \in \mathcal{A}) \vee (x \in \mathcal{A} \wedge y \in \mathcal{M})) \wedge \text{right}(x) = \text{left}(y)\} \cdot \mathcal{L}^*$ The "time" language L_2 will be designed as a standard event $L_2 = \Sigma_{\alpha\gamma\omega}$. For all decision problems α and γ remain unchanged whereas ω is adjusted appropriately.

Remark: Obviously, with little modifications L_1 can be designed as a standard event, too.

Let

$$\alpha = \{x/(x \in \mathcal{L} \cup \mathcal{M} \wedge \text{top}(x) = \emptyset) \vee (x \in \mathcal{A} \wedge \text{top}(x) = \text{in}(T)\emptyset)\}$$

and

$$\gamma = \{(x,y)/((x \in \mathcal{L} \wedge y \in \mathcal{M}) \vee (x \in \mathcal{M} \wedge y \in \mathcal{A}) \vee (x \in \mathcal{A} \wedge y \in \mathcal{M}) \vee (x \in \mathcal{A} \wedge y \in \mathcal{L})) \wedge \text{bottom}(x) = \text{top}(y)\} .$$

To show the undecidability of the CTP for regular sets we define ω in such a way that the Turing machine T stops when started with the empty tape if and only if there exists a complete (L_1,L_2)-table of size $n > 1$. Then the result follows immediately by Lemma 1.

Let

$$\omega = \{x/x \in \mathcal{L} \cup \mathcal{M} \vee (x \in \mathcal{A} \wedge \text{bottom}(x) = \text{stop}(T)a \text{ for some } a \in B_T)\}.$$

Consider a complete (L_1,L_2)-table of size $n > 1$, where

$L_2 = \Sigma_{\alpha\gamma\omega}$. Then $(RA)_i$ determines the i-th configuration of T when started with the empty tape. Furthermore, bottom$(A[n,\ell])$ = stop(T)a for some $a \in B_T$ and some $1 \leq \leq \ell \leq n$. Thus T stops. On the other hand, every accepting computation of T when started with the empty tape yields a complete (L_1,L_2)-table of size $n > 1$ (remind in(T) $\neq$ stop(T)!) because we can encode every configuration of T to a word of L_1 and control the computation by L_2. Extending the configurations by an appropriate number of $\emptyset$-symbols, we achieve always a square matrix of tiles and hence a table.

Thus we have proven the following

THEOREM 1. *The CTP is undecidable for regular sets.*

To show the undecidability of the infinite CTP for regular sets, let

$$\omega = \{x/x \in L \cup M \vee (x \in A \wedge (\text{bottom}(x) = qa \Rightarrow \Rightarrow q \neq \text{stop}(T)))\} .$$

Obviously, defining $L_2 = \Sigma_{\alpha\gamma\omega}$ for this ω, there are complete (L_1,L_2)-tables of every size, if T doesn't stop when started on the empty tape. On the other hand, if T stops within n-steps, there is no complete (L_1,L_2)-table of size greater than n. Thus we get, using again Lemma 1, the following

THEOREM 2. *The infinite CTP is undecidable for regular sets.*

We come to the last step of proof, namely to encode complete tables over arbitrary alphabets to complete tables over the two letter alphabet $\{a,b\}$. The encoding will be done in a way that single letters are represented as language tables. First we need some preparation. Consider an alphabet X and the words $w = x_1 \dots x_n \in X^*$ with $|w| = n$. We apply to these words the shift operations

$$\sigma_n^0(x_1 \dots x_n) = x_1 \dots x_n$$

and

$$\sigma_n^k(x_1 \dots x_n) = x_{n-k+1} \dots x_n x_1 \dots x_{n-k} \, (0 < k < n).$$

Define for every w, the table Ω_w of size $|w|$ by

$$\Omega_w = \begin{bmatrix} \sigma_n^0(w) \\ \vdots \\ \sigma_n^{n-1}(w) \end{bmatrix}.$$

Now let $X = \{a_1, \dots, a_m\}$. Let $p = p_1 \cdot \dots \cdot p_m$ where $p_1, \dots, p_m$ are the first m primes and let $q_i = \frac{p}{p_i}$ for every $1 \le i \le m$. Assign to every a_i the word

$$\omega_i = (ab^{p_i - 1})^{q_i} \, (1 \le i \le m)$$

and the table $\Omega_i = \Omega_{\omega_i}$ of size p.

We prove the following useful results on Ω_i:

LEMMA 2.

(1) $\forall\, 1 \le i \le m \quad \forall\, 1 \le r,s,t \le p$:

$$(\Omega_i[r,t] = \Omega_i[s,t] = a \Rightarrow (R\Omega_i)_r = (R\Omega_i)_s).$$

(2) $\forall\, 1 \le i,j \le m, i \ne j \quad \forall\, 1 \le r,s \le p \quad \exists\, 1 \le t \le p$:

$$\Omega_i[r,t] = \Omega_j[s,t] = a.$$

P r o o f. (1) is obviously true. To prove (2), consider $(R\Omega_i)_r$ and $(R\Omega_j)_s$. By definition

$$(R\Omega_i)_r = \sigma_p^k((ab^{p_i-1})^{q_i})$$

and

$$(R\Omega_j)_s = \sigma_p^\ell((ab^{p_j-1})^{q_j})$$

for some $0 < k,\ell < p$. Choose k and ℓ minimal, then $k < p_i$ and $\ell < p_j$. By this

$$(R\Omega_i)_r = b^k(ab^{p_i-1})^{q_i-1}ab^{p_i-k-1}$$

and

$$(R\Omega_j)_s = b^\ell(ab^{p_j-1})^{q_j-1}ab^{p_j-\ell-1} .$$

Consider the places where an a occurs. These are given by the formulas

$$v = k + xp_i + 1 (0 \le x < q_i)$$

and

$$v' = \ell + yp_j + 1 (0 \le y < q_j).$$

Hence we have to study the diophantine equation

$$k + xp_i + 1 = \ell + yp_j + 1 .$$

It is easily checked, because $\gcd(p_i,p_j) = 1$, that there is always a solution with $0 \le x < q_i$ and $0 \le y < q_j$, which proves the lemma.

LEMMA 3. *Every complete table A of* $\{\sigma_p^k(\omega_i)/1 \le i \le m, 0 \le k \le p\}$ *is a complete table of* $\{\sigma_p^k(\omega_j)/0 \le k < p\}$ *for some* j *; if* $(RA)_\ell, (CA)_\ell \notin \{a,b\}^*aa\{a,b\}^*$ *for* $1 \le \ell \le p$.

P r o o f. Consider ℓ with $1 \le \ell \le p$. Then $(RA)_\ell = \sigma_p^k(\omega_j) = (R\Omega_j)_{k+1}$ for some j and k and $(RA)_{\ell+1} = \sigma_p^{k'}(\omega_i) = (R\Omega_i)_{r+1}$ for some i and k'. By Lemma 2, $i = j$. Otherwise $(CA)_t \in \{a,b\}^*aa\{a,b\}^*$ for some t. Obviously the same holds for the columns. Thus there exist j,i such that

$$(RA)_\ell \in \{\sigma_p^k(\omega_j)/0 \le k < p\}$$

and

$$(CA)_\ell \in \{\sigma_p^k(\omega_i)/0 \le k < p\} \text{ for } 1 \le \ell \le p.$$

Therefore

$$\text{(i)} \quad \forall\, 1 \le r \le p \ \exists!\, 1 \le s \le p_j : A[r,s] = a$$

and

$$\text{(ii)} \quad \forall\, 1 \le s \le p \ \exists!\, 1 \le r \le p_i : A[r,s] = a.$$

Let $n = \#(\{(r,s)/A[r,s] = a \text{ and } 1 \le r \le p_i,\ 1 \le s \le p_j\})$. By (i), $n = p_i$ and by (ii), $n = p_j$. In conclusion $i = j$, completing the proof.

Consider a language $L \subseteq \{a_1,\dots,a_m\}^*$. Associate to L the language

$$L' = \bigcup_{k=1}^{p-1} \{\sigma_p^k(\omega_{i_1})\dots\sigma_p^k(\omega_{i_n})/a_{i_1}\dots a_{i_n} \in L \text{ and }$$

$$1 \le i_\lambda \le m \text{ for } 1 \le \lambda \le n\}.$$

Obviously, every complete L-table of size n transfers to a complete L′-table of size $n \cdot p$.

Now we show the converse correspondence, namely, that every complete L′-table transfers to a complete L-table. Consider a complete table A′ of L′. Then we can decompose A′ into subtables B_{rs} of size p.

$$A' = \begin{bmatrix} B_{11} & \dots & B_{1n} \\ \vdots & & \\ B_{n1} & \dots & B_{nn} \end{bmatrix}.$$

Consider an arbitrary $B_{rs} = B$. By definition of L′, $(RB)_\ell, (CB)_\ell \in \{\sigma_p^k(\omega_i)/1 \le i \le m,\ 0 \le k < p\}\setminus\{a,b\}^*aa\{a,b\}^*$ for $1 \le \ell \le p$.

Lemma 3 yields $(RB)_\ell, (CB)_\ell \in \{\sigma_p^k(\omega_{j(r,s)})/0 \le k < p\}$ for some fixed $j(r,s)$. Hence for every row R of $B_{r1}\dots B_{rn}$

$$R \in \{\sigma_p^k(\omega_{j(r,1)}\dots\sigma_p^k(\omega_{j(r,n)})/0 \le k < p\}$$

and $a_{j(r,1)}\dots a_{j(r,n)} \in L$ by definition of L′. Analogously, every column of the same "block" determines the same word in L. Thus

$$A[r,s] = a_{j(r,s)} \quad (1 \leq r,s \leq n)$$

is a complete (n,n)-table of L. In summary we have proven

THEOREM 3. *The CTP and the infinite CTP are undecidable for regular sets* $L \subseteq \{a,b\}^*$.

Remark: It is quite easy to derive the result, that the n-CTP is NP-complete for regular sets. Obviously, the problem to decide for an arbitrary non-deterministic Turing machine T and a natural number n whether or not T accepts the empty word within n steps in NP-complete. Our construction exhibits a polynomial time reduction from this problem to the n-CTP for regular sets.

REFERENCES

[1] H. BECKER and H.K.-G. WALTER, Formale Sprachen, Vieweg, 1977.

[2] U. BRANDT and H.K.-G. WALTER, The defect of language tables, Technical report, Institut für Theoretische Informatik, Fachbereich Informatik, TH-Darmstadt, 1985.

[3] R.M. ROBINSON, Undecidability and nonperiodicity for tilings of the plane, *Inventiones Math.*, 12(1971), 177-209.

[4] P. VAN EMDE BOAS, Dominoes are Forever, 1st GTI Workshop, Paderborn, 1983, 75-95.

(*Received March 6, 1986*)

Ulrike Brandt
Herman K.-G. Walter

Institut für Theoretische Informatik
Fachbereich Informatik
Technische Hochschule Darmstadt
Alexanderstrasse 24, D-6100 Darmstadt/Germany