KYBERNETIKA ČÍSLO 4, ROČNÍK 1/1965

A Full-Fledged Model of Machine Translation*

F. KIEFER, S. ABRAHAM

This paper contains the description of a semantic recognition system supposed to be used for Machine Translation based partly on the results of Chomsky's generative grammar.

As the background of the conception of this paper the reader may consult Chomsky [2], Katz and Fodor [3], and also Palek [4], and Sgall [5].

INTRODUCTION

Many problems concerning machine translation can be solved if also semantic aspects of the languages under consideration are taken into account. Up to the present time, semantic aspects can not be introduced into the programs for the analysis in the source language or for the synthesis in the target language because semantics has not been formalized. The authors of the present paper have recently outlined a formal semantic theory which makes the formal treatment of semantic features possible [1]. On the basis of the results obtained in that work we have now elaborated a new scheme of machine translation. This scheme is shown in Fig. 1 and Fig. 2.

The input of the first level of Step One consists of sentences which we assume to be grammatically correct. This assumption is not essential from the point of view of our model because if we want to test the grammaticalness of the input sentences, all we have to do is to introduce a more complete first level.

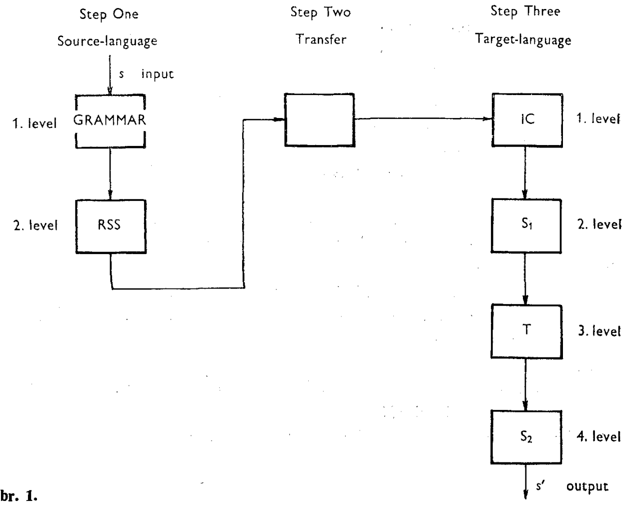
The first level has to generate a grammatical analysis of the input sentence s. As the transformational grammar is considered the most adequate grammar of language, the above=mentioned analysis is carried out on the basis of the transformational gram-

* Communicated at the *Prague Colloquium on Algebraic Linguistics and Machine Translation,* September 18th—22nd, 1964.

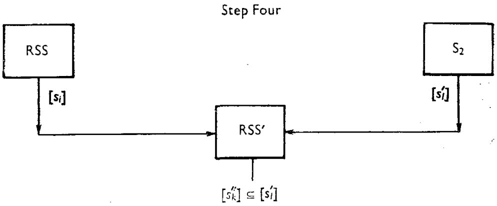
It should be made clear that the present paper gives only the rough outlines of the model and does not claim to solve all the problems involved in it. The details will be elaborated in subsequent papers.

mar. So the first level should give as output

i. the underlying kernel sentences of *s* and their *P*-markers;
ii. the ordered set of the applied transformational rules, if any.



Obr. 1.



Obr. 2.

To fulfil both these tasks, a transformational grammar must first be formulated in exact terms and preferably formulated as normal Markov-algorithms because the transformational grammar given in the form of normal Markov-algorithms, is easier for handling by computers. On the other hand, the construction of a recognition transformational grammar in the knowledge of the generative one is also possible if the latter is formulated as normal Markov-algorithms, for which the construction of inverse algorithms is known. In this way, the requirements i. and ii. can be realized.

After the grammatical analysis is carried out, the second level of Step One is put to work. The second level has to fulfil the following tasks:

i. First it has to establish whether the sentence s is semantically correct or anomalous. Of course, in this case too we could assume that the input of the second level consists only of semantically correct sentences. However, even under this assumption, we ought to be able to detect the semantic correctness of sentences because the first level may provide more than one analysis for a sentence and it is obvious that there may exist such analyses to which there correspond no correct semantic readings. In this way those grammatical analyses to which no correct semantic readings correspond are ruled out.

ii. Evidently, the words that form s may have more than one meaning. The second level has to rule out those meanings of the words which do not occur in the semantically correct readings of s.

iii. The second level has to provide for every reading such a semantic marker that enables us to check whether the sentence s', which is the translation of s, has the same semantic reading as s or not.

The input of the transfer (Step Two) consists of

i. the input-dictionary of the first level of Step One and the readings of words generated by the second level of the Step One.

Let us have the sentence s consisting of the words $w_1, w_2, \ldots, w_m$ i.e.

$$s = w_1 w_2 \ldots w_m .$$

Each w_i $(1 \leqq i \leqq m)$ can have more than one reading, i.e.

$$w_i \rightarrow \left\{ \begin{matrix} v_{w_i}^{(1)} \\ \vdots \\ v_{w_i}^{(l)} \end{matrix} \right\}$$

where l is a given positive integer. From these readings, after the second level has done its work, there remain k $(1 \leqq k \leqq l)$ readings.

ii. besides i. the input of the transfer consists of IC-rules generating the underlying kernel sentences and the set of transformational rules applied to obtain s.

More precisely, the input of the transfer is the following quadruple

$$Q = \{V_{p_i}, \Sigma_i, R_i, T_i\}$$

where V_{p_i} stands for the subset of V_p containing the words which form s with their admitted readings, Σ_i is the set of the initial strings which underlie the kernel sentences, R_i stands for the IC-rules which applied to Σ_i yield the underlying kernel sentences (more precisely, we have here a union of subsets, namely, each underlying kernel sentence has its own set of IC-rules which generate it), and T_i stands for the applied transformational rules.

The transfer has to generate a quadruple

$$Q' = \{V'_P, \Sigma'_i, R'_i, T'_i\}$$

that corresponds to Q in the following sense: namely, using V'_{P_i} and applying the R'_i rules to Σ'_i we must obtain those kernel sentences that generate after application of the transformational rules T'_i the set s'_i which we call the translation of s. The term "translation" is not defined here. We shall return to this problem and give a definition of "translation" in the discussion of Step Four.

The theory of the transfer is the real theory of translation from a scientific point of view. Moreover, the elaboration of the theory of translation will lead to a typology of languages of their transformational grammars. Since the transformational grammar completed by a semantic level is a full description of a language, a typology based on such a system is the most complete typology of all the possible ones.

Step Three has to expand the quadruple Q' on four levels. The first two levels generate all meaningful kernel sentences obtainable from Q', the third level generates the set of sentences $[s'_i]$ mentioned in the discussion of the transfer part. The fourth level provides the readings of the sentences of $[s'_i]$ by constructing their semantic markers.

Step Four yields the possibility of comparing the semantic markers of $[s]$ and those of $[s'_i]$ and on the basis of this comparison it must enable to chose those readings of $[s'_i]$ which correspond to the readings of $[s]$. Those sentences of $[s'_i]$ the semantic markers of which correspond to the semantic markers of $[s'_i]$ will be called the translations of $[s]$. (The definition of the term "translation".)

A SEMANTIC RECOGNITION SYSTEM (RSS)

The first level of Step One has to establish first the set of the underlying kernel sentences $[s^k_i]$ of s and the set of the applied transformational rules $\{T_i\}$. After this is carried out this level has to find out all the possible P-markers of these kernel sentences, more precisely, those IC-rules which, when applied to some element of Σ, yield $\{s^k_i\}$. The set of those IC-rules the application of which generates $\{s^k_i\}$ is denoted by $R^{(i)}_i$. The set of all $R^{(i)}_i$-s is denoted simply by R_i. The set of the elements to which the IC-rules are applied we denote by Σ_i.

Thus the output of the first level of Step One consists of the quadruple

$$V_{p_i}, \Sigma_i, R_i, T_i$$

where V_{p_i} stands for the set of words which form s with all its readings. By one reading of a word we understand a set of markers of the given word. For any word the number of its different readings equals the number of its different sets of markers. What is meant by a marker will be explained below.

The semantic level consists above all of a dictionary, in which each entry has the following form

$$w_i, \ (v_1^i, v_2^i, \ldots, v_j^i, \ldots)$$

where w stands for a given word and $v_1^i, v_2^i, \ldots, v_j^i, \ldots$ is a linear matrix. v_j^i is called the j-th *value of the matrix* and each value is either 1 or 0. The values of the matrices may be labelled with different names, e.g. noun, verb, being human, etc., called *categories* and then value 1 means that the given word *belongs* to the category in consideration while value 0 means that the given word does not belong to this category. The categories may be divided into two groups: the *grammatical markers* and the *semantic markers* which are needed for the formal characterization of the words. (A more detailed discussion of this topic is to be found in [1].)

Each word has at least one matrix. The matrix (matrices) belonging to a word is the reading (readings) of the word. That is, we may formulate the above statement about readings in the following way: *A word has as many readings as different matrices.*

Obviously, an unambiguous formal characterization of the words of a given language presupposes the fulfilment of the requirement that different words must have different matrices except in the case of synonymy.

Besides the dictionary the second level of Step One contains a finite set of rules, called *semantic rules* $\{s_i^r\}$ of the form

$$t_1, \ldots, t_n \rightarrow t$$

where each t_i $(1 \leqq i \leqq n)$ and t stand for a table and "→" means the replacement of the left-hand side by the right-hand side. By a table we understand a matrix in which not every square must be filled. (When filled then only with 1 or 0.)

The second level works as follows. When a sentence s arrives at this level the words of s are first replaced by the matrices of the words found in the dictionary. Thus, instead of a sequence of words (sentence) we always obtain a sequence of matrices. This sequence of matrices is called the *lexical value* of s. Obviously, each sentence may have more than one lexical value.

The rules s_i apply to the lexical values in the following way.

We shall say that a table t_i *coincides* with a matrix m_j when the following conditions are fulfilled:

a. Both are of the type $p \times q$ (i.e. both of them consist of p rows and q columns).

b. t_i and m_j have the same value (1 or 0) in the suqares in which both are filled.

Now, if we have a sequence of matrices (a subsequence of the lexical value)

$$m_1 m_2 \ldots m_r$$

and there exists such a rule S the left-hand of which coincides with it, more precisely, if each table t_i of this S rule coincides with the corresponding matrices m_j of this sequences, then this S rule may be applied to it, i.e. it can be replaced by a single matrix which must coincide with t. (The right-hand side of the S rule.)

If there exists a positive integer n such that after applying n times the rules from the set S we obtain a sequence of matrices M to which no rule S can be further applied, then we have one of the following situations:

a. M consists of one single matrix;

b. M consists of more than one matrix.

In case a. we say that the (kernel) sentece s^k is meaningful and the (single) matrix M is called the *semantic value* of s^k, and denoted by s^k_s.

In case b. or when such an n does not exist, the (kernel) sentence s^k has no meaning.

The sentence s is meaningful if and only if each of its underlying kernel sentences is meaningful.

If there exists at least one underlying kernel sentence which turns out to have no meaning then s has no meaning either and the process of the translation comes to a halt. If all underlying kernel sentences are meaningful then the procedure continues in the following way.

We may have two basic possibilities again.

i. The sentence s is obtained from one meaningful kernel sentences $s^k_1, s^k_2, \ldots, s^k_r$ by applying the transformational rules $T_1, T_2, \ldots, T_i$. In this case we define the structural meaning of s as

$$s^k_s, T_1, T_2, \ldots, T_i \, .$$

ii. The sentence s is obtained from the meaningful kernel sentences $s^k_1, s^k_2, \ldots, s^k_r$ by applying the transformational rule T. In this case we define the *structural meaning* of s as

$$(s^k_{s,1}, s^k_{s,2}, \ldots, s^k_{s,r}, T) \, .$$

All other cases are combinations of i. and ii.

In order to be able to continue the procedure we introduce three fundamental operations called *semantic operations.*

i. Θ is the operation of adding a column or a row or both to a given matrix.

ii. Π is the operation of replacing some values of the given matrix by other values.

iii. $H^{(n)}$ is a mapping of the set M into the set N where N is the set of all matrices and M the set of sequences of the form

$$s_{s_1}, s_{s_2}, \ldots, s_{s_n}$$

where s_{s_i} is a semantic value for every $1 \leqq i \leqq n$.

To each transformational rule T we assign a combination of the operations Θ and Π and denote this combinations by Γ_i.

In the case the structural meaning takes the form

$$s_{s_1}, s_{s_2}, s_{s_3}, \ldots, s_{s_n}, T$$

we apply first operation iii. and then the operation Γ_i is carried out.

In all cases mentioned above we obtain a single matrix after the application of the semantic operations. This matrix is called *distinctive semantic matrix*. Obviously, a sentence s may have more then one distinctive semantic matrix. The sentence s has as many readings as distinctive semantic matrices.

Thus, the output of Step One when operating on the sentence s is:

i. The quadruple

$$\{V_p, \Sigma_i, R_i, T_i\}$$

generated by the first level.

ii. The distinctive semantic matrix or matrices generated by the second level.

THE TRANSFER AND THE SEMANTIC GENERATIVE SYSTEM

For the quadruple

$$Q = V_{p_i}, \Sigma_i, R_i, T_i$$

the transfer yields the corresponding quadruple

$$Q' = V'_{P_i}, \Sigma_i, R_i, T'_i$$

in the target-language.

Step Three generates on the basis of Q on the first level all the possible kernel sentences using the first three sets of Q. We shall denote the set of these kernel sentences by

$$\{s_1^{k'}\} .$$

The second level rules out those kernel sentences of the set $\{s_i^{k'}\}$ that have no meaning. This is done in a similar way as in Step One. The dictionary component has to have the form

$$w_i \rightarrow m_i$$

where w_i stands for a terminal symbol (word), m_i for a linear matrix and "→" means replace w_i by m_i. These rules are called lexical rules and are denoted by $L_i^{(r)}$. The dictionary refers here to the target language.

Applying the lexical rules we obtain the *lexical value* of the sentence or sentences $s_i^{k'}$ under consideration. The rules $S_i^{(r)}$ are applied to the lexical values of $s_i^{k'}$, where the rules $S_i^{(r)}$ are built up in an analogous way as in the case of Step One (source language). If this procedure yields a single matrix the *semantic value* of the corresponding kernel sentence $s_i^{k'}$ (the kernel sentence under consideration) is meaningful, if not, it has no meaning.

So the output of the seond level of Step Three consists of

i. the semantic values of the meaningful kernel sentences generated by the first level;

ii. and the set of transformational rules $\{T'_i\}$ obtained in Step Two.

Applying the transformational rules $\{T_i'\}$ the third level generates on the basis of the meaningful sentences the set of sentences $\{s_i'\}$.

Applying the operations $\Theta, \Pi, H^{(n)}$ the fourth level generates the distinctive semantic matrices of the sentences from $\{s_i'\}$.

The output of Step Three are

i. the meaningful sentences s_i';

ii. the distinctive semantic matrices of s_i'.

Now, Step Four has as input

i. the distinctive semantic matrices of s (the output of Step One),

ii. the distinctive semantic matrices of s (the output of Step Three).

Step Four compares the distinctive semantic matrices of ii. with those of i.

The comparison is realized in the following way.

If the two distinctive semantic matrices to be compared are of the same type $p \times q$ then they will be said to be identical if they have the same value (1 or 0) in each corresponding square. If one matrix is of the type $p_1 \times q_1$, and the other of the type $p_2 \times q_2$ then we construct for each of them the matrix of the type $p' \times q'$ where

$$p' = \max(p_1, p_2)$$

and

$$q' = \max(q_1, q_2)$$

and the free squares are filled in with 0's. Then the two matrices can be compared as in the preceding case.

Those s_i'-s that have at least one distinctive semantic matrix identical with a distinctive semantic matrix of s are called the translation of s.

EXAMPLE

We take the sentence

A toll kiesett a kezéböl.

Here we have for

V_{p_i} = *kiesett, a, toll, kezéböl,*
$\Sigma_i = \{S\}$,
$R_i: S \rightarrow NPVP,\ NP \rightarrow AN,\ VP \rightarrow VNP,$
$A \rightarrow a,\ N \rightarrow toll,\ V \rightarrow kiesett,\ N \rightarrow kez\acute{e}b\ddot{o}l$, and $\{T_i\}$ is an empty set.

It should be noted that in this case the IC-rules are oversimplified because we are interested in the semantic part of the translating system only, not in the grammatical part.

Thus, the input of the second level of the first step is the triple

$$V_{p_i}, S, R_i.$$

The semantic level (second level) consists above all — as mentioned — of a dictionary component. In our case the dictionary will contain the following entries:

(1)	*a*	(1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0)
(2)	*toll*	(0 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0)
(3)	*toll*	(0 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0)

(4)	*toll*	(0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0)
(5)	*toll*	(0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1)
(6)	*toll*	(0 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0)
(7)	*kiesett*	(0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0)
(8)	*kiesett*	(0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0)
(9)	*kezéböl*	(0 1 1 0 0 0 1 1 1 1 1 1 0 0 0 0)

The categories used here for the characterizations of the words are: article, noun, concrete, verb, concrete, auxiliary, adverb, place, preposition, pronoun, possessive, object, instrument, production, part, activity. It is true, the use of 16 categories seems to be quite redundant but we must always have the union of the categories used for the characterization of words both in the source-language and the target-language. We want to stress that these categories are assumed for the sake of illustration only; they are neither sufficient for a full semantic characterization of the given words nor are they necessarily the categories at all needed for a full characterization.

We obtain the following sequences of matrices for our sentence

(1′)	(1)	(2)	(7)	(1)	(9)
(2′)	(1)	(3)	(7)	(1)	(9)
(3′)	(1)	(4)	(7)	(1)	(9)
(4′)	(1)	(5)	(7)	(1)	(9)
(5′)	(1)	(6)	(7)	(1)	(9)
(6′)	(1)	(2)	(8)	(1)	(9)
(7′)	(1)	(3)	(8)	(1)	(9)
(8′)	(1)	(4)	(8)	(1)	(9)
(9′)	(1)	(5)	(8)	(1)	(9)
(10′)	(1)	(6)	(8)	(1)	(9)

(The figures 1 through 9 in parentheses refer to the matrices given above.)

Besides, we have the S_i^r, S_i^r rules: S_1^r, S_2^r, S_3^r, S_4^r, and S_5^r:

1.
$$\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \end{array}\right)\left(\begin{array}{cccccccccccccccc} - & 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - \end{array}\right)\rightarrow$$
$$\rightarrow\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - \end{array}\right)$$

2.
$$\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \end{array}\right)\left(\begin{array}{cccccccccccccccc} - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{array}\right)\rightarrow$$
$$\rightarrow\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - \end{array}\right)$$

3.
$$\left(\begin{array}{cccccccccccccccc} - & - & - & 1 & - & - & - & - & - & - & - & - & - & - & - & - \end{array}\right)\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{array}\right)\rightarrow$$
$$\rightarrow\left(\begin{array}{cccccccccccccccc} - & - & - & 1 & - & - & - & - & - & - & - & - & - & - & - & - \\ 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{array}\right)$$

4.
$$\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & 1 & 1 & - & - & - & - & - & - & - & - & - & - & - & - & - \end{array}\right)\left(\begin{array}{cccccccccccccccc} - & - & - & 1 & 1 & - & - & - & - & - & - & - & - & - & - & - \\ 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{array}\right)\rightarrow$$
$$\rightarrow\left(\begin{array}{cccccccccccccccc} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & 1 & 1 & - & - & - & - & - & - & - & - & - & - & - & - & - \\ 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{array}\right)$$

5. $$(1\ -\ -\ -\ -\ -\ -\ -\ -\ -\ -\ -\ -\ -\ -\ -) \quad \begin{pmatrix} - & - & - & 1 & 0 & - & - & - & - & - & - & - & - & - & - & - \\ 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{pmatrix} \rightarrow$$

$$\rightarrow \begin{pmatrix} 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & 1 & 0 & - & - & - & - & - & - & - & - & - & - & - & - & - \\ 1 & - & - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ - & - & - & - & - & - & 1 & 1 & - & - & - & - & - & - & - & - \end{pmatrix}$$

First we apply the rules S_1r, S_2r, S_3r, S_4r, to (1′). So we obtain successively the sequences of matrices:

Step 1.
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} (7)\,(1)\,(9)$$

Step 2.
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} (7) \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Step 3.
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Step 4

(i)
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

So we obtain for (1′) a single matrix which we call the *semantic value* of (1′).

In a similar way we obtain single matrices for (2′), (3′), (5′) and (9′). The semantic values for these sequences are given by

(ii)
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(iii)
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(iv)
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(v)
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In the cases (4′), (6′), (7′), (8′) and (10′) we do not obtain a single matrix. Let us consider only case (4′). We apply the rules S_i^r to (4′) successively and in all possible ways:

Step. 1. After applying the rule S_1^r we obtain

$$\begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{pmatrix} (7), (1)\ (9)$$

Step 2. After applying the rule S_2^r we obtain

$$\begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix} (7) \begin{pmatrix} 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

Step 3. After applying the rule S_3^r we obtain

$$\begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \end{pmatrix} \begin{pmatrix} 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

Neither S_4^r nor S_5^r applies to this sequence.

Since our sentence is a kernel sentence and no transformational rules have been applied, the semantic values (i)—(v) are also the structural meanings and the distinctive semantic matrices of the sentence under consideration. A simple comparison shows that there is no identity between the distinctive semantic matrices, thus our sentence has five different readings.

The output of the first step is in our case

i. the triple $Q = \{V_{p_i}, S, R_i\}$ and
ii. the distinctive semantic matrices (i)—(v).

The transfer yields the quadruple $\{V'_{pi}, \Sigma'_i, R'_i, T'_i\}$ corresponding to Q' in the target language. Say, we have obtained

$V'_{pi} = \{$*the, has fallen, out, of, his, her, hand, feather, quill, pen, spring* (*of lock*), (*of key*), *bit, blade* (*of oar, scull, paddle, sweep*)$\}$

$\Sigma'_i = S$
R'_i : $S \rightarrow NPVP$
$NP \rightarrow AN$
$VP \rightarrow VNP_1$
$V \rightarrow A_nP$
$NP_1 \rightarrow P'_rNP_2$
$A \rightarrow$ *the*
$N \rightarrow$ *hand, feather, quill, pen, spring, bit, blade*
$P'_r \rightarrow$ *out*
$P'_r \rightarrow$ *of*

First Step 3. generates the following kernel sentences

1 *The hand has fallen out of his hand*
2 *The feather ...*
3 *The quill ...*
4 *The pen ...*
5 *The spring ...*
6 *The bit ...*
7 *The blade ...*
8 *The hand has fallen out of his feather*
9 *The feather ...*
10 *The quill ...*
11 *The pen ...*
12 *The spring ...*
13 *The bit ...*
14 *The blade ...*
15 *The hand has fallen out of his quill*
16 *The feather ...*
17 *The quill ...*
18 *The pen ...*
19 *The spring ...*
20 *The bit ...*
21 *The blade ...*
22 *The hand has fallen out of his pen*
23 *The feather ...*
24 *The quill ...*
25 *The pen ...*
26 *The spring ...*
27 *The bit ...*
28 *The blade ...*
29 *The hand has fallen out of his spring*
30 *The feather ...*
31 *The quill ...*
32 *The pen ...*
33 *The spring ...*
34 *The bit ...*
35 *The blade ...*
36 *The hand has fallen out of his bit*
37 *The feather ...*
38 *The quill ...*
39 *The pen ...*
40 *The spring ...*
41 *The bit ...*
42 *The blade ...*
43 *The hand has fallen out of his blade*
44 *The feather ...*
45 *The quill ...*
46 *The pen ...*
47 *The spring ...*
48 *The bit ...*
49 *The blade ...*

As mentioned, the second level of this step has to rule out those (kernel) sentences from 1—49 that have no meaning. This level consists first of the dictionary component formulated as rules (the lexical rules L^r_i).

$$W_i \rightarrow m_i .$$

Here we have the following lexical rules:

the → (1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0)
has → (0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0)
fall → (0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0)
fall → (0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0)
out → (0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0)
of → (0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0)
his → (0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0)
hand → (0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0)
feather → (0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0)
quill → (0 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0)
pen → (0 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0)
spring → (0 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0)
bit → (0 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0)
blade → (0 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0)

where the categories are the same as in the case of the source language.

With the use of the dictionary we obtain 98 sentences. As "*kiesett*" in the Hungarian sentence has also a figurative sense, we must take into account the figurative sense of the English word "*fall*". We have now 49 sentences generated by the grammatical level and in each of them "*fall*" may also occur in a figurative sense. In this way we obtain the 98 possibilities. These may be reduced by the application of the R_i-rules.

Before establishing the $R_i^{(2)}$ rules we point out the fact that the transfer specifies the $R_i^{(2)}$ rules to some extent. That means, the $R_i^{(2)}$ rules depend not only on the language they are concerned with but also on the source-language.

The $R_i^{(2)}$-rules may refer to the grammatical or to the semantic categories only or to both of them. In the case when they refer to the grammatical categories they may refer to those grammatical categories which may be in a way understood as semantic ones as well. For example, the category, "concrete" with respect to a noun may be both a semantic and a grammatical category. Thus, we have the following $R_i^{(2)}$-rules:

$$R_1^{(2)}:\quad (-\,-\,-\,1\,-\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)(-\,-\,-\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)\to$$
$$\to(-\,-\,-\,1\,-\,0\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)$$

$$R_2^{(2)}:\quad (-\,-\,-\,-\,-\,-\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)(-\,-\,-\,-\,-\,-\,-\,-\,1\,-\,-\,-\,-\,-\,-\,-\,-)\to$$
$$\to(-\,-\,-\,-\,-\,-\,-\,-\,-\,1\,1\,-\,-\,-\,-\,-\,-)(-\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)\to$$
$$\to(-\,1\,-\,-\,-\,-\,1\,1\,1\,1\,1\,-\,-\,-\,-\,-\,-)$$

$$R_3^{(2)}:\quad (1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)(-\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)\to$$
$$\to\begin{pmatrix}1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\\-\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\end{pmatrix}$$

$$R_4^{(2)}:\quad \begin{pmatrix}1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\\-\,1\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\end{pmatrix}(-\,-\,-\,1\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-)\to$$
$$\to(-\,1\,-\,-\,-\,-\,1\,1\,1\,1\,1\,0\,-\,-\,-\,-\,-)\to$$
$$\to\begin{pmatrix}1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\\-\,1\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\\-\,-\,-\,1\,1\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\,-\\-\,1\,-\,-\,-\,-\,1\,1\,1\,1\,1\,0\,-\,-\,-\,-\,-\end{pmatrix}$$

From these rules, as it may be seen, $R_1^{(2)}$ links the auxiliary verb with the main verb, $R_2^{(2)}$ links the adverb, the preposition the possessive pronoun and the noun (concrete but not an object) to an adverb of place, $R_3^{(2)}$ links the article with any noun; $R_4^{(2)}$ links a concrete noun (with article) with a concrete verb and an adverb of place; $R_2^{(2)}$ rules out those cases when the noun is concrete and the verb is figurative. As mentioned above the rule $R_2^{(2)}$ rules out all those cases in which the noun in the adverb of place is an object. This restriction, however, calls for an explanation. It is clear that the sentence

The spring has fallen out of his car

is meaningful and here "*car*" is an object. But — as we have pointed out — the $R^{(2)}$ rules are not independent of the source language. The semantic structure of the sentence in the source language imposes restrictions on the $R^{(2)}$-rules.

To take into consideration only those meanings of the words of the target language that are given by the transfer is as natural as are the restrictions imposed on the $R^{(2)}$ rules. Thus, e.g. "*spring*" has a lot of further meanings but we are interested only in the meaning "*spring of a lock*".

After the application of the $R^{(2)}$-rules only the sequence of matrices corresponding to the sentences

1′ *The feather has fallen out of his hand*
2′ *The quill ...*
3′ *The pen ...*
4′ *The spring ...*
5′ *The bit ...*
6′ *The blade ...*

remain. All these sentences are in a way synonymous. To obtain the synonymous readings of these sentences we have to consult the dictionary and to repeat the whole procedure. For these sentences we obtain the following distinctive semantic matrices:

$$1'' \quad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$2'' \quad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&0&0&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$3'' \quad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&0&0&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$4'' \quad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&1&1&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$5'' \quad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&1&1&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$6'' \quad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&1&1&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

It should be noted that 1″—6″ are originally the semantic values of the corresponding sentences. However, in the case of kernel sentences the semantic value, the structural meaning and the distinctive semantic matrices coincide. The sentences being kernel sentences, we do not need the third and fourth level of Step Three. The output of Step Three are the pairs

$$(s'_i, D'_i)$$

where s'_i stands for the generated sentences admitted by the second level, and D'_i for the corresponding distinctive semantic matrices.

Now we have to proceed with the Step Four for the comparison of the distinctive semantic matrices of the sentence s of the source language and the distinctive semantic matrices of the corresponding sentence s'_i of the target language. However, in distinctive semantic matrices is quite impossible. They are, namely, only in a few cases of the same type $m \times n$. So we need a mapping $\mathfrak{A}$ which maps the distinctive semantic matrices obtained in the source and in the target language, into the set of matrices $p \times q$. The function $\mathfrak{A}$ consists — among others — of operations of adding a row or a column to a matrix. The function $\mathfrak{A}$ is in a way determined by the transfer.

In the present case the distinctive semantic matrices of the source language have one row more than those of the target language. This row corresponds to the definite article. But we know that in English as well as in German, and some other languages, no definite article can occur before a possessive pronoun. Thus, if in the target language a possessive pronoun occurs and the target language is e.g. English, then the definite article must not be translated. In this way the transfer specifies the function $\mathfrak{A}$. For the sake of comparison the function $\mathfrak{A}$ may in this case delete the row corresponding to the definite article "*a*". In this way we obtain the following matrices of the source language:

$$\text{i}' \qquad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&1&0&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$\text{ii}' \qquad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&0&0&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$\text{iii}' \qquad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&0&0&0&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$\text{iv}' \qquad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0&0&0&0&1&1&1&1&0 \\ 0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

$$\text{v}' \qquad \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \\ 0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&1&1&1&1&0&0&0&0&0 \end{pmatrix}$$

We can now proceed to Step Four. A simple comparison shows that

i′	corresponds to	3″
ii′	corresponds to	2″
iii′	corresponds to	1″
iv′	corresponds to	4″, 5″, 6″
v′	corresponds to	

Let us consider the corresponding sentences:

A toll kiesett a kezéböl with the distinctive semantic matrix i′ corresponds to *The pen has fallen out of his hand.*

A toll kiesett a kezéböl with the distinctive semantic matrix ii′ corresponds to *The quill has fallen out of his hand.*

A toll kiesett a kezéböl with the distinctive semantic matrix iii′ corresponds to *The feather has fallen out of his hand.*

A toll kiesett a kezéből with the distinctive semantic matrix iv′ has three correspondences, namely *The spring has fallen out of his hand, The bit has fallen out of his hand, The blade has fallen out of his hand* where *spring* is here understood as *spring of lock, bit* as *bit of key* and *blade* as *blade of oar, blade of scull, blade of paddle, blade of sweep.*

A toll kiesett a kezéből with the distinctive semantic matrix v′ strictly has no correspondence in English. As can be seen in this case both "*toll*" (*pen*) and "*kiesett*" (*has fallen*) are meant in a figurative sense. The Hungarian sentence has the idiomatic meaning 'The writer died". If we give the idioms in a list we may get a suitable translation for our sentence with the distinctive semantic matrix v′, namely, for instance,

He died.

In this way we have obtained all the translations of the Hungarian sentence

A toll kiesett a kezéből.

Another problem is of course, how to find out the suitable translation of this sentence, if it occurs in a context. In our case the sentence has been regarded separately with no restrictions imposed on the meaning. The next problem we want to scrutinize is now the restrictions imposed on the source language sentence can be transfered to the corresponding sentence (*s*) in the target-language.

(Received January 12th, 1965.)

REFERENCES

[1] F. Kiefer, S. Abraham: The Theory of Structural Semantics. To be published by Mouton & Co. The Hague.
[2] N. Chomsky: Syntactic Structures. The Hague 1964.
[3] J. J. Katz, J. A. Fodor: The Structure of a Semantic Theory. Language. 39 (1963), 170–210.
[4] B. Palek: Informace o transformační gramatice. Slovo a slovesnost 24 (1963), 140—151.
[5] P. Sgall: Generativní systémy v lingvistice. Slovo a slovesnost 25 (1964), 274—282.

Zlepšený model strojového překladu

F. Kiefer, S. Abraham

Stať se opírá o teoretické výsledky širší práce připravené k tisku [1]. Je tu rozvíjen návrh sématického popisu vět založený z části na Chomského principech generativních gramatik. Návrh sledující potřeby strojového překladu obsahuje čtyři kroky. Prvý se týká gramatického a sémantického popisu věty vstupního jazyka. Druhý je převodem do jazyka výstupního. Třetí krok se zabývá sestavením vět výstupního jazyka na čtyřech rovinách, z nichž poslední podává významy vět konstruováním jejich sémantických vzorců. Ve čtvrtém kroku je prováděn výběr právě jednoho významu věty. V závěru je postup ilustrován informativní ukázkou.

Na základě použitého matematického postupu je pak dodána definice pojmu a termínu „překlad“.

Ferenc Kiefer, Samuila Abraham, Computing Centre of the Hungarian Academy of Sciences. Uri utca 53, Budapest I. Hungary.