50X1-HUM

Page Denied

Next 2 Page(s) In Document Denied

Declassified in Part - Sanitized Copy Approved for Release 2014/03/04 : CIA-RDP80-00247A003100530001-7

Applications of graph theory to mathematical logic and linguistics

K. Čulík

First of all, graphs whose vertices are words, i.e. elements of a free semigroup, will be considered. Then some labelled graphs will be mentioned; and an example of a double graph, i.e. of a set with two binary relations will be exhibited. Finally, ordinary undirected graphs will also be treated.

1. Mathematical logic and theory of languages

The main aim of mathematical logic is the study of the binary relation of deducibility in various logical formalisms or abstract languages. From a quite general point of view the situation may be described as follows.

A finite alphabet (or vocabulary) V is given; the universal language V^{∞} is the free semigroup of strings (or words or sentences) over V under the operation of concatenation with the identity element $J \in V$. Further, a finite set of proper rules $\mathcal{R} \subset V^{\infty} \times V^{\infty}$ is given, i.e. we have a graph $\langle V^{\infty}, \mathcal{R} \rangle$ the vertices or edges of which are strings or proper rules resp. This graph contains only a finite number of edges. Using a typical context operator $\mathcal{C}$ we assign to the finite relation $\mathcal{R}$ a uniquely determined infinite binary relation

(1) $C\mathcal{R} = \{(p,q)$; there are $x, y, u, v \in V^{\infty}$ such that $(u,v) \in \mathcal{R}$ and $p = xuy,\ q = xvy\}$.

Now, in the infinite graph $\langle V^{\infty}, C\mathcal{R} \rangle$ the finite paths $(w_1, w_2, \dots, w_k)$, $k \geq 2$ (i.e. $(w_i, w_{i+1}) \in C\mathcal{R}$ for each

$i = 1, 2, \ldots, k-1$) are said to be derivations (or proofs). If we denote by T the transitive closure operator defined for all binary relations, then $T C \mathcal{R}$ is essentially the binary relation of deducibility. If we take a set $A \subset V^{\infty}$ as the set of axioms then the set

(2) $S(A) = \{x$; there is a $y \in A$ such that $(y, x) \in T C \mathcal{R}\}$

is the set of all theorems which may be deduced from A by a constructive mapping S called a semialgorithm. This notion is intimately related to the semi-Thue combinatorial systems of M. Davis [11] and to the associative calculi of A. A. Markov [12].

One of the main aims of mathematics is to characterize the infinite classes by a structure using finite classes only. Here, the infinite binary relations $T C \mathcal{R}$ and $C \mathcal{R}$ should by characterize by the properties of the finite relation $\mathcal{R}$. For these purposes it is useful to investigate what properties of binary relations are invariant in regard to the operators C, T and TC. If a property of $\mathcal{R}$ is not invariant, it is necessary to find the corresponding property of $C\mathcal{R}$ or $TC\mathcal{R}$; and also conversely. I shall show some examples of the questions of this type.

If $\mathcal{R} \neq \emptyset$, there are infinitely many relations $\mathcal{R}_i$ such that $\mathcal{R}_i \subset \mathcal{R}_{i+1} \neq \mathcal{R}_i$ and $C\mathcal{R}_i = C\mathcal{R}$ for each $i = 1, 2, \ldots$ Sometimes it is possible to omit some elements of $\mathcal{R}$ without any changes of $C\mathcal{R}$. Therefore we say that $\mathcal{R}$ is C-irreducible if there is no relations $\mathcal{R}^*$ such that $\mathcal{R}^* \subset \mathcal{R} \neq \mathcal{R}^*$ and $C\mathcal{R}^* = C\mathcal{R}$. If D denotes the operator

(3) $D\mathcal{R} = \{(u, v) \in \mathcal{R}$; there are no $x, y \in V^{\infty}$ such that $xy \neq J$ and $(xuy, xvy) \in \mathcal{R}\}$,

then the following theorem is proved easily.

Theorem 1. An arbitrary binary relation $\mathcal{R}$ is C-irreducible if and only if $\mathcal{R} = D\mathcal{R}$. Further more, $CDC\mathcal{R} = C\mathcal{R}$ and $DCD\mathcal{R} = D\mathcal{R}$.

If we want to investigate the class of binary relation $C\mathcal{R}$ only, it suffices to restrict to to the class of binary relations, $D\mathcal{R}$ and in this case D is an operator inverse to C. From this point of view the following condition is important

(4) if $(p, q) \in C\mathcal{R}$, there are uniquely determined $x, y \in V^{\infty}$ and $(u, v) \in \mathcal{R}$ such that $p = xuy$, $q = xvy$.

Theorem 2. An arbitrary binary relation $\mathcal{R}$ satisfies condition (4) if and only if $\mathcal{R}$ is C-irreducible and $\mathcal{R}$ satisfies (5). If $\mathcal{R}$ satisfies (6), then $\mathcal{R}$ is C-irreducible and satisfies (4).

(5) There are no pairs $(u_1, v_1), (u_2, v_2) \in \mathcal{R}$ such that $(u_1, v_1) \neq (u_2, v_2)$ and $xu_1 = u_2 y$, $xv_1 = v_2 y$ for some $x, y \in V^{\infty}$.

(6) If $(u, v) \in \mathcal{R}$ and $u = xu'$, $v = yv'$ or $u = u'x$, $v = v'y$ where $xy \neq y$, then $x \neq y$.

In a similar manner the operators T and TC may be treated. We say that $\mathcal{R}$ is TC-irreducible if there is no $\mathcal{R}^*$ such that $\mathcal{R}^* \subset \mathcal{R} \neq \mathcal{R}^*$ and $TC\mathcal{R}^* = TC\mathcal{R}$.

Theorem 3. An arbitrary binary relation $\mathcal{R}$ is TC-irreducible if and only if $\mathcal{R}$ is C-irreducible and if $\mathcal{R}$ satisfies the following condition (7). If $\mathcal{R}$ is TC-irreducible then $\mathcal{R}$ satisfies condition (8).

(7) If $(u,v)\in\mathcal{R}$ then there is no path $(u=w_1,\dots,w_k=v)$ in $C\mathcal{R}$ such that $k>2$ and $(w_i,w_{i+1})\neq(u,v)$ for all i, $1\leq i<k$.

(8) If $(u_i,v_i)\in\mathcal{R}$ for $i=1,2$, then $(u_1u_2,\ v_1v_2)\notin\mathcal{R}$.

If is clear that condition (7) is unsuitable because it concerns the infinite relation $C\mathcal{R}$. On the other hand (7) is very similar to the condition of atransitivity. A binary relation $\mathcal{R}$ is said to be <u>atransitive</u> if it satisfies

(9) if $(u,v)\in\mathcal{R}$, then $u\neq v$ and there is no path $(u=w_1,\dots,w_k=v)$ in $\mathcal{R}$ such that $k>2$.

If $\mathcal{R}$ is an atransitive binary relation then $C\mathcal{R}$ need not be atransitive, i.e. the property of atransitivity is not invariant under the context operator C. E.g., let $\mathcal{R}$ contain the three pairs (u_i,v_i) for $i=1,2,3$, where $v_2=cv_3d$, $u_1=acu_3db$ and $v_1=au_2b$. Now by (1), $(au_2b,\ acv_3db)\in C\mathcal{R}$, $(acu_3db,\ acv_3db)\in C\mathcal{R}$, and of course also $(acu_3db,\ av_2b)\in C\mathcal{R}$; from (9) it follows that $C\mathcal{R}$ is not atransitive. The condition for $\mathcal{R}$ which corresponds to atransitivity of $C\mathcal{R}$ seems rather complicated.

Originally I was led to these questions by investigations of the axiomatic system for <u>phrase structure grammars</u> of N. Chomsky [4] where the axioms and the restrictions express some properties of $C\mathcal{R}$. It was suitable to reformulate these axioms so as to concern the finite relation $\mathcal{R}$ only. It was shown [9] that a new axiom must be added in order to obtain an equivalent axiomatic system.

2. Syntactic structure of sentences

There are two important modifications of semialgorithms: a) the alphabet V is decomposed into two disjoint parts T and N which are said to be the terminal (proper) and nonterminal (auxiliary) alphabets ($J \in T$) ; and b) if $(u, v) \in \mathcal{R}$, then $u \in N$, i.e. u is a single letter. In this case the semialgorithm is determined by a more special condition

(2') $S(A) = T^{\infty} \cap \{ x \;;$ there is a $y \in A$ such that $(y, x) \in TC\mathcal{R} \}$.

This semialgorithm S and the languages $S(A)$, i.e. the sets of strings generated by it, are called context-free [6]. A very important example is the programming language ALGOL 60 [10], where the elements of N or T are said to be the metalinguistic variables or basic symbols respectively, and $\mathcal{R}$ contains about 150 proper rules which are determined by the syntactic definitions. Then for program $\in N$, the language S (≺program≻) is the set of all programs written in ALGOL 60.

In automatic programming it is necessary to recognize the syntactic structure of each particular program $p \in S$ (≺program≻) in order to translate it into the computer language. The syntactic structure of p is determined by a derivation $(w_1, \ldots, w_k)$ such that $w_k = p$ and $w_1 =$ ≺program≻; unfortunately however, there often are several derivations, distinct as sequences of strings, which determine the same syntactic structure. A necessary equivalence relation concerning the set of derivations may be introduced using the notion of isomorphism of labelled double graphs as follows.

If $(w_1, w_2, \ldots, w_k)$ is the given derivation, we may write $w_i = a_{i,1} a_{i,2} \ldots a_{i,m_i}$, where $a_{i,j} \in V$ and

$v_{i,j} \neq J$ for all $i, 1 \le i \le k$, and all $j, 1 \le j \le m_i$ (i.e. m_i is the length of the string w_i). From the fact that $(w_i, w_{i+1}) \in C\mathcal{R}$ and that the considered semialgorithm is context-free there follows the existence and unicity of indices d_i and $d_i + e_i \ge 1$ such that $(v_{i,d_i}, v_{i+1,d_i} \dots v_{i+1, d_i + e_i}) \in \mathcal{R}$ for $i = 1, 2, \dots, k-1$ (i.e. the condition (4) is always satisfied here, but under the assumption that $(u,v) \in \mathcal{R}$ implies $u \neq v$.

Now in the set $M = \{(i,j);\ 1 \le i \le k,\ 1 \le j \le m_i\}$ of <u>places</u> we introduce three binary relations ϱ, π, σ as follows:

1) If $(i,j), (r,s) \in M$ then $(i,j)\varrho(r,s)$ if $r = i+1$ and $s = j$ for $j < d_i$, but $s = j + e_i$ for $j > d_i$ for all $i = 1, 2, \dots, k-1$. If $\bar{\varrho}$ denotes the smallest equivalence relation defined in M and containing ϱ, let $[i,j]$ denote the equivalence class containing (i,j) and let $\bar{M}$ be the set of all equivalence classes. Inductively one may prove

<u>Lemma 1.</u> $[i,j] = [i+p, s]$, where $1 \le i \le k,\ 0 \le p \le k-i,\ i \le s \le m_{i+p}$ is valid if and only if

$s = j + \sum_{q=1}^{\ell} e_{a_q}$ for some e_{a_q}, where $i \le a_1 < a_2 < \dots < a_\ell \le i+p$

and if the following inequalities are satisfied

$j < d_m$ for each m with $i < m < a_1$, $j > d_{a_1}$;

for each $t, 1 \le t \le \ell$; $j + \sum_{q=1}^{t} e_{a_q} < d_m$ for each

$m, a_t < m < a_{t+1}$, $j + \sum_{q=1}^{t} e_{a_q} > d_{a_{t+1}}$,

$s < d_m$ for each $m, a_\ell < m \le i+p$.

If we start from $i=1$ and $j=1,2,\dots,m_1$ we get from Lemma 1 the description of identification of places. We may introduce a mapping f as follows: $f(i,j)=n_{ij}$ for $1\le i\le k$, $1\le j\le m_i$; then from $(i,j)\varrho(r,s)$ follows $f(i,j)=f(r,s)$ and therefore it is possible to define a labelling $\bar f$ by the condition $\bar f[i,j]=f(i,j)$.

2) If $(i,j),(r,s)\in M$ then $(i,j)\pi(r,s)$ if $r=i+1$, $j=d_i$ and $d_i\le s\le d_i+e_i$. It is clear that $\pi\cap\varrho=\emptyset$ and that in each equivalence class $[i,j]$ only, for the places with maximal or minimal first index denoted by $(i_{max},p)\in[i,j]$ or $(i_{min},q)\in[i,j]$, may the following condition be valid: $(i_{min},q)\pi(r,s)$ or $(r,s)\pi(i_{max},q)$ fore some $(r,s)\in M$. Therefore it is possible to extend the relation $\bar\pi$ from M to $\bar M$ by the following definition: $[i,j]\bar\pi[r,s]$ if there are $(i_{min},p)\in[i,j]$ and $(r_{max},q)\in[r,s]$ such that $(i_{min},p)\pi(r_{max},q)$.

Lemma 2. Each of the connected components of the graph $\langle\bar M,\bar\pi\rangle$ is a rooted directed tree with the root $[1,j]$ for some j, $1\le j\le m_1$, i.e. the vertex $[r,s]\in\bar M$ has input degree 0 or 1 according as it is or not the root.

3) If $(i,j),(r,s)\in M$ then $(i,j)\sigma(r,s)$ if $r=i$ and $s=j+1$. It is clear that $\pi\cap\sigma=\varrho\cap\sigma=\emptyset$ and that the graph $\langle M,\sigma\rangle$ consists of k simple paths. From this and from the definition of equivalence classes $[i,j]$ it follows that it is possible to define a relation $\bar\sigma$ as follows: $[i,j]\bar\sigma[r,s]$ if $(p,q)\sigma(v,w)$ for some $(p,q)\in[i,j]$ and $(v,w)\in[r,s]$ ✝. A binary relation R is said to be

✝ or if there are $[a_k,b_k]$ such that $[a_1,b_1]\bar\pi[a_2,b_2]\bar\pi\dots\bar\pi[a_n,b_n]$, $n\ge 2$ and either $[a_1,b_1]=[r,s]$ and $[i,j]\bar\sigma[a_n,b_n]$ or $[a_n,b_n]=[r,s]$ and $[i,j]\bar\sigma[a_1,b_1]$.

acyclic, if it satisfies the following condition

(10) if $(u, v) \in \mathcal{R}$, then $u \neq v$ and there is no path $(v = w_1, \ldots, w_n = u)$ in $\mathcal{R}$.

Furthermore, from the known properties of rooted trees it follows that there is an uniquely determined path which connects an arbitrary vertex $[i, j] \in \bar{M}$ with the root in $\langle \bar{M}, \bar{\pi} \rangle$ and which is called rooted path for $[i, j]$.

Inductively one may prove easily

Lemma 3. In the directed graph $\langle \bar{M}, \bar{\sigma} \rangle$ the relation $\bar{\sigma}$ is atransitive and acyclic and the input or output degree of the vertex $[i, j] \in \bar{M}$ is 0 if and only if $j = 1$ or $j = m_i$ resp. If $([i, j] = [i_1, j_1], \ldots, [i_p, j_p] = \text{root})$ and $([h, k] = [h_1, k_1], \ldots, [h_q, k_q] = \text{root})$ are rooted paths in $\langle \bar{M}, \bar{\pi} \rangle$ for $[i, j]$ and $[h, k]$ resp., then $[i, j] \, \bar{\sigma} \, [h, k]$ if and only if $[i, j] \neq [h, k]$ and either $k_q = j_p + 1$ or $h_q = j_p$, but $k_{q-t} = j_{p-t} + 1$ for the maximal index t such that $k_{q-v} = j_{p-v}$ for all $v = 0, 1, \ldots, t-1$.

Using the previous lemmas it is possible to prove the following

Theorem 4. Both the relations $T\bar{\pi}$ and $T\bar{\sigma}$ are partial orderings, i.e. asymmetric and transitive, and in $T\bar{\pi}$ there is at most one chain connecting two vertices. Further $T\bar{\pi}$ and $T\bar{\sigma}$ satisfy the following condition:

(11) if $x, y \in \bar{M}$ and $x \neq y$, then there always occurs precisely one of the following possibilities: $(x, y) \in T\bar{\pi}$, $(x, y) \in T\bar{\sigma}$, $(y, x) \in T\bar{\pi}$ and $(y, x) \in T\bar{\sigma}$;

(12) if $(x, y) \in T\bar{\sigma}$ and $(x, x'), (y, y') \in T\bar{\pi}$, then

$(x', y'), (x, y'), (x', y) \in T\bar{\sigma}$.

The labelled double graph $\langle \bar{M}, \bar{\Gamma}, \bar{\bar{\pi}}, \bar{\bar{\sigma}} \rangle$, i.e. the set with labelled elements and with two binary relations, is said to be <u>phrase marker</u> [5] of the string w_k which is determined by the given derivation $(w_1, w_2, \ldots, w_k)$ in the given contex-free semialgorithm (in this case $m_1 = 1$). A maximal path in regard to $\bar{\bar{\sigma}}$ is called the <u>structure index of</u> w_k, which is necessary for the transformational grammars [5].

Two phrase markers are <u>isomorphic</u> if they are isomorphic in regard to both relations and also to the labelling. Two different derivations of the same string determine the same syntactic structure if their phrase markers are isomorphic.

The Lemmas 1-3 show that the decision problem concerning the equivalence or nonequivalence of two given derivations is a rather complicated procedure even for a computer.

Of course the relation $\bar{\bar{\pi}}$ or $T\bar{\bar{\pi}}$ is the <u>generative subordination</u> and $\bar{\bar{\sigma}}$ or $T\bar{\bar{\sigma}}$ the <u>word order relation</u>.

A similar construction of the phrase marker in a general semialgorithm (not necessarily context-free) leads to a more general double graph. In this case $T\bar{\bar{\pi}}$ is a quite general partial ordering (not satisfying the given specification, and in (12) it is necessary to suppose $x' \neq y'$). Therefore a set together with two partial orderings which satisfy (11) and (12) is suitable for the description of the syntactic structure of an arbitrary sentence.

<u>3. Finite automata and switching theory</u>

Each mapping is a binary relation and, together with its arguments and values, it may be considered as a directed graph. This was really done by the many authors who treated automata and switching theory. This representation is very suitable in the case of complicated superpositions of many mappings. Always the graphs thus introduced are labelled. I give here only some examples.

Let $G = \langle M, E, f \rangle$ be a finite directed and labelled multigraph and let L be a fixed finite set of labels, i.e. $f(e) \in L$ for each egde $e \in E$. A path in G is a finite sequence of edges $(e_1, e_2, \ldots, e_m)$ such that output vertex of e_i is the same as the input vertex of e_{i+1} for $i = 1, 2, \ldots, m-1$. We say that this path generates the string $f(e_1) f(e_2) \ldots f(e_m) \in L^\infty$. Let $G(X, Y)$, where $X, Y \subset M$, be the set of all strings generated by all paths starting at a vertex from X and ending in a vertex from Y. The following was proved in [8]

Theorem 5. The class of all sets $G(X, Y)$ for arbitrary G, X, Y is the set of all regular events in an alphabet L in the sense of S. C. Kleene [14].

These finite directed and labelled multigraphs are strongly connected with a special type of context-free semialgorithm. The special condition is the following: if $(u, v) \in R$ then $v = tw$, where $t \in T$ and $w \in N$ or $w = J$ and we put $T = L$, $N = M$. In this case the vertices are said to be states and the proper rules transition rules in the finite state grammar (essentially) G [3].

On the other hand, let the labelling f of G satisfy the following condition: if $e_1, e_2 \in E$, $e_1 \neq e_2$ and e_1, e_2 have the same input vertex, then $f(e_1) \neq f(e_2)$. Further let g be another labelling of vertice, i.e. $g(v) \in K$ for each $v \in M$, where K is a finite set the elements of which are called outputs (and the elements of L inputs). Finally, let one vertex $v_0 \in M$ be distinguished as the initial state. Under these definitions the double labelled graph $\langle M, E, f, g \rangle$ is the graph of the finite automaton of E. F. Moore [14]. If we choose the second labelling in a different manner, we get the graph of the finite automaton of G. H. Mealy [13]. Using other types of labellings we get logical nets of A. W. Burks and J. B. Wrigt [2], etc.

In these fields there arise many special questions concerning graph theory from a new point of view.

4. Semantics and translation

Let L be the set of lexical units of several natural languages and let $\rho \subset L \times L$ be the binary relation of translatability [1]. We may suppose that ρ is symmetric (both L and ρ may be found in ordinary dictionaries concerning the investigated languages). Further let M be a set whose elements are called meanings.

A multivalued mapping φ of L into M is said to be a semantic, if φ satisfies the following conditions

(13) $\emptyset \neq \varphi(x) \subset M$ for each $x \in L$,

(14) $(x, y) \in \rho$ if and only if $\varphi(x) \cap \varphi(y) \neq \emptyset$,

(15) if $p \in M$ then there is at least one $x \in L$ such that $p \in \varphi(x)$.

Condition (14) says that x may be translated as y if x and y have at least one common meaning.

Now we put $\mathfrak{M} = \{\varphi(x);\ x \in L\}$ and define a binary relation $\mu \subset M \times M$ as follows: $(P, Q) \in \mu$ if $P, Q \in \mathfrak{M}$ and $P \cap Q \neq \emptyset$. The graph $\langle \mathfrak{M}, \mu \rangle$ is a set-theoretical representation in M of the graph $\langle L, \rho \rangle$, because φ is a strong homomorphism which maps $\langle L, \rho \rangle$ onto $\langle \mathfrak{M}, \mu \rangle$. Further let $\mathcal{Y}$ be a set of complete subgraphs in $\langle L, \rho \rangle$ which satisfies the following condition:

(16) each $x \in L$ and each $(y, z) \in \rho$ is contained in some subgraph $\langle V, \sigma \rangle \in \mathcal{Y}$.

In this case we say that $\mathcal{Y}$ covers $\langle L, \rho \rangle$.

The smallest cardinal number of $\mathcal{Y}$ covering $\langle L, \rho \rangle$ is said to be the number of completeness of $\langle L, \rho \rangle$ and denoted by $\omega\langle L, \rho \rangle$. It is easy to prove

Lemma 4. $\omega\langle L, \rho \rangle$ is equal to the smallest cardinal number of M such that M admits a set theoretical representation of $\langle L, \rho \rangle$, i.e. such that there exists a mapping φ satisfying (13) - (15).

Furthermore, it is easily shown that to each set $\mathcal{Y}$ satisfying (16) there may be found another set $\mathcal{Y}'$ of complete subgraphs of $\langle L, \rho \rangle$ such that

(17) card $\mathcal{Y}' \leq$ card $\mathcal{Y}$,

(18) if $\langle V', \sigma' \rangle \in \mathcal{Y}'$ then there is a subgraph $\langle V, \sigma \rangle \in \mathcal{Y}$ such that $V' \subset V$, $\sigma' \subset \sigma$,

(19) the set $\mathcal{W}'$ of all different sets V' such that $\langle V', \sigma' \rangle \in \mathcal{Y}'$ is a decomposition of L.

From (19) it follows that $\mathcal{W}'$ is a chromatic decomposition [7] of the graph $\langle L, L \times L - \rho \rangle$, i.e. if $x, y \in V'$, $V' \in \mathcal{W}'$ then $(x, y) \notin L \times L - \rho$. If $\chi\langle L, \rho \rangle$ denotes the chromatic number of $\langle L, \rho \rangle$ one may prove the following

Theorem 6 $\chi\langle L, L \times L - \rho \rangle \leq \omega\langle L, \rho \rangle$.

Similar difficulties arise with the number of completeness as with the chromatic number.

The number of completeness $\omega\langle L, \rho\rangle$ is the smallest number of different meanings of the given system of translation $\langle L, \rho\rangle$.

References

[1] Andreev N. D., Fitialov S. Ja.: Interlingua of machine translation and principles of its construction. Abstracts of conference in mathematical linguistics, Leningrad 1959, 53-60.

[2] Burks A. W., Wright J. B.: Theory of logical nets. Proc. IRE (1953), 41, 1357-1365.

[3] Chomsky N., Miller G. A.: Finite state languages. Inf. and Control (1958), 1, 91-112.

[4] Chomsky N.: On certain formal properties of grammars. Inf. and Control (1959), 2, 137-167.

[5] Chomsky N.: On the notion „rule of grammar". Prof. of S.M. (1961), vol. XII (Structure of language and its mathematical aspects), 6-24.

[6] Chomsky N.: A note on phrase structure grammars. Inf. and Control (1959), 2, 393-395.

[7] Čulík K.: On chromatic decompositions and chromatic numbers of graphs, Publ. Fac. Sci. Univ. Brno, Tchécoslovaquie, No. 403 (1959), 177-185.

[8] Čulík K.: Some notes on finite state languages and events represented by finite automata using labelled graphs. Čas. pro pěst. mat. (1961), 86, 43-55.

[9] Čulík K.: On some axiomatic systems for formal grammars and languages. Proc. IFIP Congress 62, North-Holland 1963.

[10] Čulík K.: Formal structure of ALGOL and simplication of its description. Symbolic languages in data prosessing, Gordon and Breach 1963, 75-82.

[11] Davis M.: Computability and Unsolvability, McGraw-Hill 1958.

[12] Марков А. А.: Теория алгорифмов. Труды мат. инст. В. А. Стеклова XLII, Изд. АН СССР 1954.

[13] Mealy G. H.: Method for synthesizing sequential circuits. Bell syst. techn. jour. (1955), 34, 1045-1079.

[14] Автоматы, Изд. иностр. лит., Москва 1956.

Adresa:

Matematický ústav ČSAV, Praha 1, Žitná 25, ČSSR.

Semantics and translation of grammars and ALGOL - like languages

K. Čulík, Prague

The language L is determined by the set of expressions E and by the semantisc S. E contains certain strings of basic symbols choosen from V_T and S is a function which joins a meaning $S(x)$ to each expression $x \in E$.

The translation from the language L into the language L' is the function T' such that $T(x) \in E'$ and $S(x) = S'(T(x))$ for each $x \in E$. The problem is how to determine the translation T in some economical and constructive way (not as the set of couples $x, T(x)$ for all $x \in E$). This may be done if we use suitable grammars generating the given languages.

The set E is generated by the context-free grammar $G = \langle V_T, V_N, \mathcal{R} \rangle$ if $E = \bigcup_{x \in V_N} G(x)$, where $G(x)$ denotes the set of all expressions $e \in E$ such that there is a derivation in G beginning with x and ending with e. V_N contains the metalinguistic variables and $\mathcal{R}$ syntactical definitions (or rules) $\mathcal{V} = a_0 ::= a_1 a_2 \ldots a_n$; where $a_0 \in V_N$ and $a_i \in V_N \cup V_T$ for each $i = 1, 2, \ldots, n$.

In regard to the translation T the set V_T is divided in two parts V_A and V_P. V_A contains auxiliary symbols (as brackets, space, coma and other separators) and V_P the proper symbols (the symbols in V_P usually are again the strings over an alphabet A_P). Each rule $\mathcal{V} \in \mathcal{R}$ can be unically expressed in the standard form $a_0 ::= b_0 a_1 b_1 \ldots a_k b_k$, where b_i are either zero-strings or strings over V_A and $a_i \in V_P \cup V_N$; a_i is said to be the i-th <u>standard symbol</u> of $\mathcal{W}$.

Further we define an extension $\tilde{S}$ of S as follows: $\tilde{S}(x) = S(x)$ for $x \in E$ and $\tilde{S}(x) = \bigcup_{y \in G(x)} \tilde{S}(y)$ for $x \in V_N$.

The grammar G of the language L (i.e. G generates E of L) is said to be <u>well translatable</u> into the grammar G' of L' if there is a mapping Φ of $\mathcal{R}$ into $\mathcal{R}'$ and another mapping τ of V_P into V_P' and of V_N into V_N' satisfying the following conditions:

(1) if $a_0 ::= b_0 a_1 b_1 \ldots a_n b_n$ and $c_0 ::= d_0 c_1 d_1 \ldots c_m d_m$ are the standard forms of the rules $\mathcal{W}$ and $\Phi(\mathcal{W})$ resp. then $m = n$ and there is a permutation π of the set $\{1, 2, \ldots, n\}$ such that $c_{\pi(i)} = \tau(a_i)$ for each $i = 1, 2, \ldots, n$ and

(2) if $x_i \in G(a_i)$ and $y_i \in G'(c_i)$ and $S(x_i) = S'(y_{\pi(i)})$ for all $i = 1, 2, \ldots, n$ then $S(b_0 x_1 b_1 \ldots x_n b_n) = S'(d_0 y_1 d_1 \ldots y_n d_n)$.

If G is well translatable into G' we obtain the expression $T(x) \in E'$ from the expression $x \in E$ using three algorithms (A), (B) and (C) as follows:

(A) <u>Analysis</u>

1. We begin with $x_0 = x$ and we make the 1-st step (x_0 does not contain any symbol from V_N).

2. In the i-th step we deal with x_{i-1} and we take a rule $\mathcal{W} \in \mathcal{R}$ (it is necessary to search it) such that $x_{i-1} =$ $= v b_0 a_1 b_1 \ldots a_k b_k w$, where v and w are some suitable strings and $a_0 ::= b_0 a_1 b_1 \ldots a_k b_k$ is the standard form of $\mathcal{W}$. We construct $x_i = v a_0 w$ and we put $r(a_0) = i$, where the integer $r(a_0)$ is called the <u>rang of metalinguistic variable</u> a_0 in x_i . The rangs of metalinguistic variables contained in v or w in x_{i-1} remain unchanged also in x_i. We store the statements "the i-th rule is $\mathcal{W}$" as the pair (i, $\mathcal{W}$) and "h-th rule was used to the j-th standard symbol

of the i-th rule" as the triple (h,j,i) if $a_j \in V_N$ and $r(a_j) = h$ in x_{i-1}.

3. We stop if $x_n \in V_N$.

The result of the analysis is a set P of pairs $(i, \mathscr{W})$ and a set Q of triples (h,j,i). It is clear that by P and Q is determined the phrase marker of x in G.

(B) Translation of phrase marker

We construct $P' = \{(i, \Phi(\mathscr{W})); (i, \mathscr{W}) \in P\}$ and $Q' = \{(h, \pi_i(j), i); (h,j,i) \in Q\}$, where π_i is the permutation belonging in Φ to the i-th rule (i.e. to the rule $\mathscr{W}$ such that $(i, \mathscr{W}) \in P$). It is clear that by P' and Q' is determined a phrase marker in G' of the required expression $x' = T(x)$.

(C) Synthesis

1. We begin with $x'_n = \mathscr{T}(x_n)$, where x_n is the last string in our analysis, we define $r(x'_n) = n$ and we make the 1-st step.

2. In the i-th step we deal with x'_{n-i+1} and we take the rule $\mathscr{W}'$ such that $(i, \mathscr{W}') \in P'$. We concern $c_0 \in V'_N$ such that $x'_{n-i+1} = vc_0w$ for some strings v and w and $r(c_0) = i$. We construct $x'_{n-i} = vd_0c_1d_1 \ldots c_kd_kw$, where $c_0 ::= d_0c_1d_1 \ldots c_kd_k$ is the standard form of $\mathscr{W}'$ and we define $r(c_j) = h$ in x'_{n-1} if $(h,j,i) \in Q'$. The range of other metalinguistic variables rimains unchanged again.

3. We stop with x'_0 and put $x' = x'_0$.

May 5th 1964

Adress: Mathematical Institute of ČSAV, Prague 1, Žitná 25, Czechoslovakia.

On equivalent and similar grammars of ALGOL - like languages

K. Čulík, Prague

Let $G = \langle T, N, \mathcal{R}, S \rangle$ be a context-free grammar, i.e. T and N are terminal and nonterminal vocabularies resp., $S \in N$ and in $\mathcal{R}$ are the rules $(a_0, a_1 a_2 \dots a_m)$, where $a_0 \in N$, $a_i \in T \cup N$ for each $1 \leq i \leq m$, $m \geq 1$ ($a_1 a_2 \dots a_m$ is said to be string over $T \cup N$). E.g. in ALGOL 60 T and N are sets of basic symbols and metalinguistic variables resp., S = < programm > and $\mathcal{R}$ contains elementary syntactic definitions x : : = yzm (the metasymbol $|$ meaning "or" is omitted). Let L be the language generated by G and let $\mathcal{P}$ be the set of all phrase markers of elements in L. The phrase markers were intriduced by N. Chomsky and in [1] are defined as double graphs the vertices of which are labelled by symbols of $T \cup N$.

If we identify two nonterminal symbols x and y, i.e. if we substitute x instead of y in all places in all rules of $\mathcal{R}$ and if we omit y from N, we get a new grammar G^*. It is easy to see that $L^* \supset L$ and the mapping Φ of $\mathcal{P}$ into $\mathcal{P}^*$ is determined by the mentioned substitution. If Φ is a mapping onto $\mathcal{P}^*$ then x and y are said to be interchangeable in G. E.g. if to each $(a_0, a_1 a_2 \dots a_m) \in \mathcal{R}$ where $a_0 = x$ (or $a_0 = y$) exists another rule $(b_0, b_1 b_2 \dots b_m) \in \mathcal{R}$ such that $m = m$, $b_0 = y$ (or $b_0 = x$) and for each i, $1 \leq i \leq m$ either $a_i = b_i$ or $a_i, b_i \in \{x, y\}$, then x and y are interchangeable.

The homomorphism of a grammar G_1 onto a grammar G_2 is a mapping φ of $T_1 \cup N_1$ onto $T_2 \cup N_2$ such that 1) φ is an one-to-one mapping of T_1 onto T_2, 2) $(a_0, a_1 a_2 \dots a_m) \in \mathcal{R}_1$

implies $(\varphi(a_0), \varphi(a_1)\varphi(a_2)\cdots\varphi(a_m)) \in \mathcal{R}_2$ and 3) φ induces the mapping Φ of $\mathcal{P}_1$ onto $\mathcal{P}_2$. Two grammars are said to be equivalent if it is possible to map them homomorphically onto the same grammar.

Another way how to change the grammars are extensions and reductions. A grammar G_p is an extension of the grammar G_0 if there are grammars $G_1, G_2, \ldots, G_{p-1}$ such that the following condition holds for each i, $1 \le i \le p$: there exists a rule $(a_0, a_1 a_2 \ldots a_m) \in \mathcal{R}_{i-1}$, a symbol $b \notin \bigcup_{t=0}^{i-1} (N_t \cup T_t)$, an index j, $1 \le j \le m$ and an integer $k \ge 0$, $1 \le j + k \le m$ such that $T_i = T_{i-1}$, $N_i = N_{i-1} \cup \{b\}$, $\mathcal{R}_i = (\mathcal{R}_{i-1} - \{(a_0, a_1 a_2 \ldots a_m)\}) \cup \{(a_0, a_1 \ldots a_{j-1} b a_{j+k+1} \ldots a_m)(b, a_j a_{j+1} \ldots a_{j+k})\}$ and $S_i = S_{i-1}$. E.g. it may be $\mathcal{R}_0 = \{(S, brt), (S, bw)\}$ and $\mathcal{R}_2 = \{(S, by), (y, w), (S, bx), (x, rt)\}$. In this case x and y are interchangeable, but they do not satisfy the above mentioned sufficient condition.

The composition + of two sets of rules $\mathcal{X}_1$ and $\mathcal{X}_2$ is defined as follows: $\mathcal{X}_1 + \mathcal{X}_2 = \{(a_0, x_0 y_1 x_1 \ldots y_m x_m);\ x_i$ and y_j are some strings such that there are $(a_0, x_0 b_1 x_1 \ldots b_m x_m) \in \mathcal{X}_1$ and $(b_j, y_j) \in \mathcal{X}_2$ for some symbols b_j and for each j, $1 \le j \le m\}$.

A nonterminal symbol x of the grammar G is said to be reductible if there is no rule in $\mathcal{R}$ of the form (x, p), where p is a string containing x. The symbol x is reductible if and only if in $\mathcal{X}_1 + \mathcal{X}_2$ is no rule containing x, where $\mathcal{X}_1$ and $\mathcal{X}_2$ are the sets of all rules in $\mathcal{R}$ containing x in their right and left side resp.

Let x be a nonterminal reductible symbol in G. It is natural to construct a new grammar G^* as follows: $N^* = N - \{x\}$

and $\mathcal{R}^* = (\mathcal{R} \cup (\mathcal{T}_1 + \mathcal{T}_2)) - (\mathcal{T}_1 \cup \mathcal{T}_2)$ G^* is said to be direct reduction of G. A grammar G_p is called reduction of the grammar G_0 if there are $G_1, G_2, \dots G_{p-1}$ such that G_i is direct reduction of G_{i-1} for each i, $1 \leq i \leq p$. Some simple examples of the reduction in ALGOL 60 are shown in [2].

Now two grammars are said to be strong or week similar if they have equivalent extensions or reductions resp.

If x and y are interchangeable in G and G^* is direct reduction of G with the reduced symbol z, $x \neq z \neq y$, then x and y are interchangeable in G^* again. If two grammars are strong similar then they are week similar too, but not conversally. E.g. $\mathcal{R}_1 = \{(S, bz), (b, xy)\}$ and $\mathcal{R}_2 = \{(S, xc), (c, yz)\}$ are week similar because $\mathcal{R} = \{(S, xyz)\}$ is their common reduction, but there are evidentely no equivalent extensions of them. There are some lattice properties of the greatest extensions and smallest reductions in regard to the equivalence relation among the grammars.

[1] K. Čulík: Applications of graph theory to mathematical logic and linguistics, Theory of graphs and its applications,

[2] K. Čulík: Formal structure of ALGOL and simplication of its description, 75-82, Symbolic languages in data processing, (Roma 1962), Gordon-Breach, N.Y. 1963.

Adress: Mathematical Institute of Czech. Acad. of Sciences
Prague, Jan. 28. th, 1964.