KYBERNETIKA ČÍSLO 2, ROČNÍK 2/1966

Ein mehrstufiges generatives System*

PETR SGALL

Es werden kurz und ohne Formalisierung graphentheoretische Mittel angedeutet, die zur Beschreibung der Satzstruktur dienen können. Eine Möglichkeit wird besprochen, wie diese Mittel (die projektiven Bäume) linearisiert werden können; sie werden damit in Formeln abgebildet, die mittels Kellerspeichertransduktoren bearbeitet werden können, um aus der Repräsentierung des Satzes auf der Ebene der semantischen Satzstruktur seine Repräsentierung auf der Ebene der Satzglieder zu bekommen. Ein Programm mathematischer Bearbeitung dieser Linearisierung der projektiven Bäume wird kurz angedeutet.

1.

In der Stellung der algebraischen Linguistik kann man ein Paradox sehen: Einerseits können wir hier als klar voraussetzen, daß die algebraische Linguistik keine spezielle Disziplin ist, daß sie eher die Grundlage der modernen theoretischen Linguistik im Ganzen bildet. Andererseits gibt es aber auch heute nur sehr wenige Linguisten, die in der algebraischen Linguistik tätig sind. Der neue, mathematische Standpunkt und die mit ihm verknüpften methodologischen Erfordernisse finden immer noch keine breite Annahme außer kleinen Gruppen von Spezialisten. Für die Verknüpfung der formalen Theorie mit der empirischen Bearbeitung verschiedener natürlicher Sprachen ist aber die Beteiligung größerer Kollektive notwendig. Es scheint also auch aus diesen Gründen – neben anderen – wichtig, die Arbeit in der neuen Richtung womöglich eng an die Ergebnisse der klassischen Linguistik anzuknüpfen.

Man sollte also unter anderem solche mathematische Mittel für die Sprachbeschreibung suchen, deren Berechtigung von den meisten Linguisten ohne weiteres anerkannt werden müßte und die nicht als etwas „nichtlinguistiches“ angesehen werden könnten.

* Vortrag am *Prager Kolloquium über die algebraische Linguistik und Maschienenübersetzung*, 18.—22. September 1964.

Wenn wir solche anschaulichere, üblichere Mittel suchen, die in der Linguistik schon längst benützt werden, da bieten sich uns vor allem die von den klassischen und tradizionellen syntaktischen Arbeiten bekannten Diagramme, die im Rahmen der Graphentheorie ohne große Schwierigkeiten bearbeitet werden können.

Es handelt sich uns hier nicht um Derivationsbäume, die aus den Grammatiken Chomskys bekannt sind, sondern um Abhängigkeitsbäume. Die Diagramme z. B., die in der Syntax der tschechischen Sprache üblich sind [1], können ohne grundsätzliche Schwierigkeiten in Abhängigkeitsbäume übergeführt werden, mit denen D. G. Hays und andere arbeiten [2]. Es gibt hier Unterschiede in der unterliegenden linguistischen Konzeption (z. B. die Relation vom Subjekt zum Prädikat wird in einem Abhängigkeitsbaum als eine Abhängigkeitsrelation gedeutet [3]).

Wir setzen also voraus, daß zur Beschreibung der Struktur des Satzes ein doppelter orientierter Graph mit den folgenden Eigenschaften dienen kann: 1. eine von den beiden Relationen des Graphen, die Relation R, ist eine solche, daß der Graph ein Wurzelbaum ist; die zweite, Relation W, definiert auf der Menge seiner Knotenpunkte eine vollständige Anordnung [4]; 2. der Baum ist projektiv, d. h. wenn $R(A, B)$ für zwei seine Knotenpunkte A und B gilt und ein dritter Knotenpunkt C existiert, wobei $W(A, C)$ und auch $W(C, B)$ – oder $W(B, C)$ und auch $W(C, A)$ – gelten, dann gilt immer auch $R(C, B)$ [5; 6]. In dem entsprechenden Positionsdiagramm wird die Relation W so angedeutet, daß A immer mehr links als B steht, wenn $W(A, B)$ gilt; die Relation R so, daß A immer unter B gestellt und mit ihm (nicht immer unmittelbar) mittels einer Linie verbunden wird, wenn $R(A, B)$ gilt. Die Knotenpunkte und Kanten sind bewertet. Wir interpretieren die Bewertungen der Knotenpunkte als Wörter und die Bewertungen der Kanten als die syntaktischen Funktionen; $R(A, B)$ wird als „A hängt von B ab“ interpretiert (wo die Abhängigkeit unmittelbar oder auch durch andere Wörter vermittelt werden kann); $W(A, B)$ interpretieren wir als „A steht vor B“.

2.

Die linguistische Konzeption, die wir mit diesem graphentheoretischen Apparat bearbeiten möchten, geht davon aus, daß dem Satz nicht nur *eine* syntaktische Struktur eigen ist, sondern daß es da mehrere Ebenen gibt, von denen wir mindestens die des sog. semantischen Satzbaus [7; 8] und die der üblichen Satzglieder bearbeiten wollen, neben der morphologischen und der „niedrigeren“ Ebenen (die letzten werden in diesem Aufsatz nicht besprochen). (Das Verhältnis der semantischen Satzstruktur und der Satzgliederebene kann man mit der Beziehung von Chomskys Tiefenstruktur und Oberflächenstruktur vergleichen, nicht aber in allen Hinsichten.) Wir wollen zwischen den einzelnen Ebenen immer eine Beziehung von Funktion und Form (in der Terminologie der Prager linguistischen Schule) sehen, d. h. die Reihenfolge der Ebenen soll eine Zergliederung der Beziehung vom Inhalt zum Ausdruck in der Sprache darstellen. (Mathematisch kann die Beziehung von Funktion und Form alls eine allgemeine Abbildung einer Ebene auf die ihr nächste Ebene angesehen werden, wenn wir jede Ebene als eine bestimmte Menge von Repräsentierungen der Sätze charakterisieren.) Die Ebene der semantischen Satzstruktur hat ihre eigenen syntaktischen Beziehungen, wo Relationen wie Agens, Patiens, freie Determination wichtige Rolle spielen. (Im Weiteren schreiben wir dafür R_a, R_p, R_d.) Dagegen auf der Ebene der Satzglieder bestehen Beziehungen wie Subjekt, Objekt, Attribut, Adverbialbestimmung. Es soll zuerst die Repräsentierung des Satzes auf der Ebene des semantischen Satzbaus generiert werden, die dann von der einen Ebene auf die andere übergeführt wird, bis wir zur phonetischen (oder graphischen) Repräsentierung kommen. Man sollte allerdings nicht nur bei dem Satz als der größten Einheit bleiben, sondern zu Einheiten wie Aussage (Mitteilung, Text) übergehen; das ist aber heute nur ein Programm.

3.

Die Repräsentierung des Satzes auf der Ebene der semantischen Satzstruktur kann in der Form eines Graphen des erwähnten Typs dargestellt werden (vgl. das Beispiel im Bild 1). Den Übergang zur Ebene der Satzglieder können wir in der Terminologie der Graphentheorie z. B. so beschreiben: Wir gehen von der Wurzel des Baumes aus (d. h. vom Prädikat des Satzes, bzw. des Hauptsatzes, was die linguistische Interpretation betrifft). Wir folgen der linksten Kante, immer weiter, bis wir zu einem Endpunkt des Graphen kommen (wo es keine weitere Kante mehr gibt). Von hier kehren wir der letzten bisher behandelten Kante entlang wieder zurück, bis wir zu

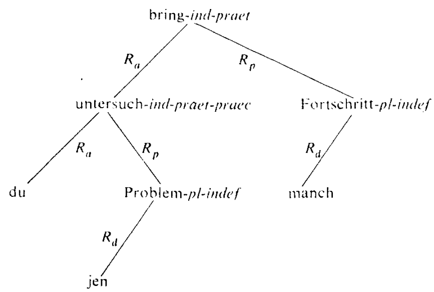


Bild 1. Struktur des Satzes *„Deine Untersuchung jener Probleme brachte manche Fortschritte.“* Die „Suffixe“ bei den einzelnen Wörtern sind als „Bedeutungen“ grammatischer Morpheme gedacht (Indikativ, Präteritum, Vorzeitigkeit, Unbestimmtheit, Plural usw.); vgl. dazu [11].

einem solchen Knotenpunkt kommen, der mit einer noch freien Kante (d.h. mit einer, der wir noch nicht gefolgt sind) inzident ist. In unserem Beispiel (Bild 1) ist es zunächst der mit *untersuch ...* bewertete Knotenpunkt. Dann gehen wir wieder nach unten, dieser freien Kante (bzw. der linksten von den bisher freien Kangen) folgend, usw. Schließlich kommen wir dann zu einem Kontenpunkt, der mit keiner freien Kante mehr inzident ist und von dem wir auch nicht mehr nach oben einer Kante entlang gehen können. In solchem Fall sind wir schon allen Kanten hin- und zurückgefolgt und sind jetzt zur Wurzel zurückgekehrt. Dieser ganze Prozeß kann mittels eines einfachen Algorithmus beschrieben werden.

	Fortsetzung nach der Antwort JA	Fortsetzung nach der Antwort NEIN
(1) Beginne bei der Wurzel des Baumes!	(2)	—
(2) Ist dieser Knoten mit einer noch freien Kante inzident?	(3)	(4)
(3) Gehe nach unten, der linksten noch freien Kante entlang!	(2)	—
(4) Ist dieser Knoten mit einer von ihm nach oben führenden Kante inzident?	(5)	Ende
(5) Gehe nach oben, dieser Kante entlang!	(2)	—

Während dieses Prozesses sollen wir auch verschiedene Operationen durchführen, womit die Bewertung der Knoten und Kanten mittels einer anderen ersetzt wird (d.h. wir ersetzen die Symbole der Ebene der semantischen Satzstruktur durch diejenigen der Ebene der Satzglieder). Jedesmal bei der ersten Berührung eines Punktes des Baumes ersetzen wir den Semoglyph (z. B. „arbeit-") durch eine der entsprechenden Wortbedeutungen (z. B. *Arbeit*, *arbeiten*, *arbeitend*). Es ist nicht notwendig tatsächlich alle Symbole zu verändern; es gibt eine bestimmte Hierarchie der Repräsentationsrelation [9] und man kann – um bei dem einfachsten Beispiel zu bleiben – in unserem Fall das Verb *arbeiten* mit demselben Symbol bezeichnen wie den Semoglyph „arbeit-", da diese Beziehung als primär gelten kann, während *Arbeit* und *arbeitend* auf der Ebene der Satzglieder spezielle Symbole haben müssen. Ähnlich ist es auch bei der Bewertung der Kanten.

Diese Entsprechung der Symbole kann mittels Tabellen gegeben werden (vgl. [8]). Es können bei diesem Vorgang die Bedingungen respektiert werden, mit denen man in der klassischen Linguistik am häufigsten arbeitet; sie müssen allerdings empirisch weiter studiert werden. Man kann diese Bedingungen, die beim Überführen des Satzes von einer höheren Ebene auf die nächste untere entscheidend sind, in die folgenden vier Gruppen einteilen:

(a) Semantik und Wortart des betreffenden Wortes;

(b) seine sog. syntaktische Funktion – d.h. die Art seiner Beziehung zum regierenden Wort;

(c) Semantik und syntaktische Funktion verschiedener von ihm abhängiger Wörter;

(d) morphologische oder lexikalische Eigenschaften des regierenden Wortes;

(e) stylistische Bedingungen.

Die Auswahl der Ausdrucksmittel nach verschiedenen stylistischen Bedingungen
spiegelt sich in unserem System durch eine freie Auswahlsmöglichkeit an einigen Stellen der Tabellen; die Bedingungen dieser Auswahl werden nicht formuliert. Die Bedingungen (a), (b) können grundsätzlich mittels einer Tabelle formuliert werden (wie kompliziert sie sein muß, ist eine andere Frage). Die Möglichkeit einer Formulierung von (c) und (d) wird dadurch gesichert, daß wir bei der Veränderung der Bewertung eines Knotens in so einem Punkt des oben charakterisierten Prozesses sind, wo die unter ihm gelegenen Knoten noch unberührt sind (d.h. die von ihm abhängigen Wörter haben noch die Form der höheren Ebene, vgl. Punkt (c)) und die über ihm gelegenen Knoten haben ihre Bewertung schon in der verwandelten Form (d.h. in der Form der unteren Ebene, vgl. (d)).

Für diese Operationen genügt es also, Tabellen zu haben, nach denen wir immer vor der Durchführung der Regel (2) die Bewertung des Knotens (und, wenn wir nicht von (5) kommen, auch die der soeben durchgegangenen Kante) verändern können. Daneben gibt es auch Fälle, wo zwei Knotenbewertungen ihren Platz wechseln sollen (vgl. z. B. russ. пять часов, wo das Zahlwort auf der Ebene der Semantischen Satzstruktur als vom Substantiv abhängig gilt, auf der Ebene der Satzglieder aber als sein Regierungswort), oder wo eine neue Kante mit neuem Endknoten gebildet wird (für Modalverba, Konjunktionen usw.). Weiter müssen auch Beziehungen wie Koordination und Apposition beschrieben werden. Das alles kann (mindestens beim jetzigen Zustand der Forschung) mehr genau im automatentheoretischen Apparat beschrieben werden, als in dem bisher erwähnten graphentheoretischen (vgl. dazu weiter unten).

In einer ähnlichen Weise kann dann der Satz von der Ebene der Satzglieder zur morphologischen Ebene übergeführt werden, wobei die Kongruenz- und Rektionskategorien, sowie Präpositionen generiert werden. Hier werden auch die Abweichungen der Wortfolge von den mit der Projektivität verbundenen Regeln beschrieben: z. B. eine Konstruktion wie *ein besserer Redner als Forscher* wird auf der Ebene der Satzglieder (sowie auf derjenigen der semantischen Satzstruktur) in der Weise behandelt, als wäre die Wortfolge *ein Redner besserer als Forscher*; erst beim Übergang zur morphologischen Ebene, für die die Projektivität nicht mehr wesentlich ist, und wo die Abhängigkeitsverhältnisse verdunkelt werden können, werden *besserer* und *Redner* umgetauscht.

4.

Wenn wir das oben kurz charakterisierte System genauer formulieren wollen, um z. B. seine generative Stärke mit derjenigen der bekannten Typen von Grammatiken und Automaten vergleichen zu können, brauchen wir schon andere Mittel. Es ist dann notwendig, lineare Formeln zu finden, die den oben beschriebenen projektiven Bäumen ein-eindeutig zugeordnet werden können. Es können Formeln sein, in denen den Kantenbewertungen des Graphen Funktoren und den Knotenbewertungen elementare Argumente entsprechen. Eine Möglichkeit solcher Linearisierung kann durch folgende Regeln definiert werden:

	JA	NEIN
(1) Beginne bei der Wurzel des Baumes und schreibe ihre Bewertung ab!	(2)	–
(2) Ist dieser Knoten mit einer noch freien Kante inzident?	(3)	(4)
(3) Gehe nach unten, der linksten noch freien Kante entlang; schreibe die Bewertung dieser Kante ab (mit einem zugefügten Strich, wenn die Kante nach rechts geht); schreibe die Bewertung des nun erreichten Knotens ab!	(2)	–
(4) Ist dieser Knoten mit einer nach oben führenden Kante inzident?	(5)	Ende
(5) Schreibe eine linke Klammer und kehre dann der Kante entlang zum nächsten oberen Knoten zurück!	(2)	–

Wenn beim Durchführen dieser Regeln das Resultat stets von rechts nach links geschrieben wird, steht in der Formel immer das Symbol, das dem regierenden Wort entspricht, rechts von allen von ihm abhängigen Wörtern (Wortstellung „regens post rectum“); wenn im Satz mehrere Wörter von demselben regierenden Wort abhängig sind, ist die Anordnung der ihnen entsprechenden Symbole in der Formel ihrer Wortfolge (wie sie aus dem Baum ersichtlich ist) entgegengesetzt. Zum Beispiel dem Baum vom Bild 1 entspricht jetzt die Formel (I):

(I) ((manch R_d Fortschritt-*pl-indef* R'_p (((jen R_d Problem-*pl-indef* R'_p (du R_a untersuch-*ind-praet-praec* R_a bring-*ind-praet*

Man kann feststellen, daß dieser Algorithmus eine beiderseits eindeutige Abbildung der Menge der im § 1 charakterisierten projektiven Bäume auf eine Menge der Formeln definiert. Das ist wohl – ohne einen Beweis – aus der folgenden Erwägung ersichtlich (wo $\{A_i\}$ die endliche Menge der Knotenbewertungen des Baumes ist, $\{R_j\}$ die endliche Menge seiner Kantenbewertungen):

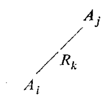
Setzen wir eine Sprache L voraus, die als eine Menge von Formeln mittels einer rekursiven Definition charakterisiert werden kann:

1. jedes Element der Menge $\{A_i\}$ ist ein Element der Menge L';
2. jedes Element der Menge L' ist ein Element der Menge L (eine Formel der Sprache L);
3. wenn x ein element von L' und y ein Element von L ist, dann ist $(yR_jx$ ein Element von L';
4. wenn x und auch y ein Element von L ist, ist auch $(yR'_jx$ ein Element von L;
5. L hat keine anderen Elemente als die von 2. und 4., L' keine anderen als die von 1. und 3.

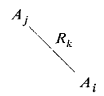
a) Unser Algorithmus bildet jeden projektiven Baum in eine Formel der Sprache L ab: Den Baum mit einem einzigen Knotenpunkt, dessen Bewertung A_i ist, bildet er in die Formel A_i ab; wenn ein projektiver Baum mehr als einen Knotenpunkt hat, kann man ihn in zwei projektive Bäume X, Y teilen, und zwar so, daß wir die erste von den mit der Wurzel des Baumes inzidenten Kanten, von der rechten Seite zählend, streichen; wenn die gestrichene Kante von der Wurzel zur rechten Seite gerichtet war, kommen wir mittels des Algorithmus zur Formel $(yR'_jx$ (wo x und y die den Bäumen X, Y mittels unserem Algorithmus zugeschriebenen Formeln der Sprache L sind, R_j die Bewertung der gestrichenen Kante; wir nehmen an, daß X von den beiden Bäumen der ist, zu dem die Wurzel des ursprünglichen Baumes gehört); wenn die gestrichene Kante von der Wurzel zur linken Seite gerichtet war, kommen wir zur Formel $(yR_jx$. Die projektiven Bäume X und Y können, wenn sie mehr als einen Knotenpunkt haben, in derselben Weise weiter geteilt werden, usw., bis wir zu Bäumen von je einem Knotenpunkt kommen. – Daß der Algorithmus jedem projektiven Baum nur eine einzige Formel zuschreibt, geht daraus hervor, daß er ein Algorithmus ist (ohne Alternativen in der Wahl einzelner Schritte).

b) Zwei verschiedene projektive Bäume werden in zwei verschiedene Formeln abgebildet: Wenn die beiden projektiven Bäume verschiedene Anzahl von Knotenpunkten oder verschiedene Bewertung von Knotenpunkten und Kanten haben, ist das klar. Sonst gibt es aber bei den projektiven Bäumen nur solche Unterschiede, die den folgenden Beispielen entsprechen (oder auch Kombinationen dieser Fälle):

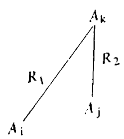
projektiver Baum — entsprechende Formel



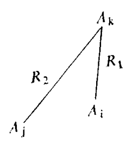
$(A_i\ R_k\ A_j$



$(A_i\ R'_k\ A_j$



$A_j\ R_2\ (A_i\ R_1\ A_k$



$A_i\ R_1\ (A_j\ R_2\ A_k$

(und ähnlich mit den nach rechts orientierten Kanten)

c) Jede Formel der Sprache L wird mit unserem Algorithmus einem projektiven Baum zugeschrieben, denn die Formel A_i wird dem Baum von einem mit A_i bewerteten Knoten zugeschrieben, und die Formel $(xR'_iy$, wo x, y Formeln der Sprache L sind, wird dem projektiven Baum zugeschrieben, der in die den Formeln x und y entsprechenden Bäume geteilt werden kann (vgl. sub a), ähnliches gilt für die Formel $(xR_iy$. Andere Formeln enthält die Sprache L nicht (vgl. ihre Definition).

Diese Linearisierung des Baumes ermöglicht es, die Überführung der Repräsentierung des Satzes von der einen auf die andere Ebene in der Form eines Kellerspeicherautomaten (Eveys pushdown store transducer) zu formulieren, denn die Sprache L ist (bis auf unwichtige Einzelheiten der Notation) der Sprache L_1^0 gleich, die wir in [8] charakterisiert haben und die (wie dort angedeutet wurde) mittels solcher Automaten in weitere Ebenen übergeführt werden kann. Wenn der Automat von rechts nach links liest und von links nach rechts druckt (womit wir auch im Weiteren rechnen), wird dabei die Anordnung „regens post rectum" erhalten, während die Anordnung der von demselben Wort unmittelbar abhängigen Wörter (Symbole) mit jedem Durchgang umgekehrt wird.

Die Anordnung „regens post rectum", zu der unser Algorithmus führt, brauchen wir, um die im § 3 als (a)–(d) bezeichneten Bedingungen mittels eines Kellerspeicherautomaten durchführen zu können. Wenn wir z. B. die Kongruenzkategorien oder andere Angaben vom regierenden Wort zu den von ihm abhängigen Wörtern übertragen sollen, kann das bei dieser Anordnung verhältnismäßig einfach formuliert werden; denn das dem regierenden Wort entsprechende Symbol kommt vor allen von ihm abhängigen in den Speicher, und wenn ein Symbol im Input gelesen wird, ist sein „regierendes" Symbol immer am zugänglichen Ende des Speichers, sodaß seine verschiedenen Eigenschaften bei der Bearbeitung der abhängigen Symbole respektiert werden können.

Von der Formel (I) kommen wir so – wie in [8] ausführlicher gezeigt wurde – zur Formel (II), d.h. zur linearen Repräsentierung desselben Satzes auf der Ebene der Satzglieder:

(II) ((du R_{at} ((jen R_{at} Problem-*pl-indef* R'_{at} Untersuchung R_{sb} ((manch R_{at} Fortschritt-*pl-indef* R'_{ob} bring-*ind-praet*

Diese Formel (wo die Funktoren als Attribut, Subjekt und Objekt interpretiert werden) kann auch als Ergebnis der Anwendung eines ähnlichen Algorithmus auf den projektiven Baum vom Bild 2 angesehen werden. Und zu diesem Baum kommen wir, wenn wir den Baum vom Bild 1 in der im § 3 angedeuteten Weise bearbeiten. Wir können also den Baum vom Bild 1 als die Repräsentierung unseres Satzes auf der Ebene der semantischen Satzstruktur ansehen – ähnlich wie die Formel (I), und den Baum vom Bild 2 als seine Repräsentierung auf der Satzgliederebene.

Eine Formel wie (II) kann dann – beim Übergang zur morphologischen Ebene – mittels eines Automaten des erwähnten Typs zur terminalen Wortfolge übergeführt werden. Ähnlich kann auch der projektive Baum (z. B. der vom Bild 2) zur morphologischen Repräsentierung des Satzes übergeführt werden (vgl. § 3), wobei die Kongruenz- und Rektionskategorien gewonnen werden und die Abhängigkeitsrelation nicht mehr mit übergetragen wird.

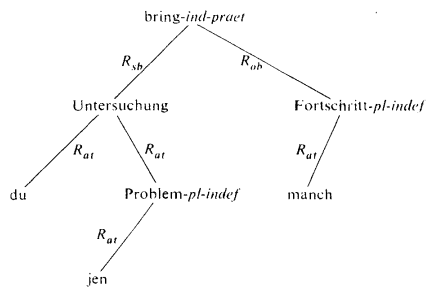


Bild 2. Repräsentierung des erwähnten Satzes auf der Satzgliederebene.

Wenn es also gelingt, natürliche Sprachen mittels der erwähnten Graphen und Tabellen zu beschreiben, dann könnte man ein mathematisch verhältnismäßig einfaches System gewinnen, das nicht nur die Sprachen generiert und ihren Sätzen befriedigende Strukturbeschreibungen zuordnet, sondern auch eine klare und von der klassischen Linguistik nicht viel entfernte linguistische Interpretation hat.

Auch die Beziehung des generativen Systems zu einem Modell des Sprechenden oder Hörenden wird klarer, wenn wir mit einem solchen mehrstufigen System arbeiten, vgl. [10]. In mancher Hinsicht scheint dieses System auch für die automatische Übersetzung gut geeignet zu sein: Es kann darin mindestens ein Teil der (lexikalischen, nicht nur grammatischen) Semantik beschrieben werden; wenn die „Ellipsenregeln" erst in den Übergang von der morphologischen zur phonologischen Ebene eingegliedert werden, so existiert eine Identifikationsprozedur, die für die Analyse notwendig ist.

(Eingegangen am 2. Dezember 1964.)

LITERATUR

[1] Vl. Šmilauer: Novočeská skladba. Praha 1947.

[2] D. G. Hays: Dependency Theory: A Formalism and Some Observations. Language *40* (1964), 511—525.

[3] S. J. Fitialov: О моделировании синтаксиса в структурной лингвистике. Проблемы структурной лингвистики, Москва 1962, 100—114.

[4] K. Čulík: O popisech větné struktury (in: N. D. Andrejev: Studie z aplikované lingvistiky. Praha 1963).

[5] Y. Lecerf: Programme des conflits, modèle des conflits. La traduction automatique *1* (1960), Nr. 4, 11ff; Nr. 5, 17ff.

[6] S. Marcus: Sur la notion de projectivité. Computational Linguistics *III* (1964), 75ff.
[7] P. Sgall: Zur Frage der Ebenen im Sprachsystem. Travaux linguistiques de Prague *1* (1964), 95—106.
[8] P. Sgall: Generative Beschreibung und die Ebenen des Sprachsystems (vorgelesen am Symposion Zeichen und System der Sprache II, Magdeburg 1964).
[9] J. Kuryłowicz: Заметки о значении слова. Вопросы языкознания *4* (1955), № 3, 73—81.
[10] P. Sgall: Generation, Production and Translation (vorgelesen an der Konferenz über Maschinenübersetzung in New York, 1965).
[11] P. Sgall: The Intermediate Language in MT. Computational Linguistics *II* (1963), 35ff.

VÝTAH

Několikastupňový generativní systém

PETR SGALL

V článku se ukazuje, jak by bylo možné v algebraické lingvistice bez velkého odchýlení od dosavadních lingvistických zvyklostí užívat některých prostředků teorie grafů. Syntaktická stavba věty může být zachycena v podobě orientovaného stromu s vrcholem, kde jsou uzly i hrany ohodnoceny a kde jsou definovány dvě relace (závislostní — odpovídající směru hran, a slovosledná — odpovídající postavení jednoho uzlu vlevo nebo vpravo od druhého). Ohodnocení hran odpovídá syntaktickým vztahům (považujeme-li přísudek hlavní věty za jediný nezávislý člen), ohodnocení uzlů — lexikálním prvkům.

Přechod od jedné roviny k druhé při derivaci věty v generativním systému může být popsán jako určitý postup, při němž procházíme celý strom a podle tabulek měníme ohodnocení jednotlivých uzlů a hran, popř. i (ve zvlášť vymezených případech) hrany obměňujeme nebo dodáváme.

Ukazuje se také možnost, jak uvedené prostředky srovnat co do generativní síly se známými typy gramatik a automatů; toto srovnání se děje s pomocí několikastupňového generativního systému založeného na zásobníkovém automatu. Stručně se naznačuje program matematického zpracování tohoto způsobu linearizace grafů uvedeného typu.

Doc. dr. Petr Sgall, CSc., Centrum numerické matematiky mat.-fyz. fakulty KU, Malostranské nám. 25, Praha 1