ÜBER DIE REKURSIVITÄT DER BEGRIFFE DER MATHEMATISCHEN GRAMMATIKEN

von

RÓZSA PÉTER

Einleitung

1. Die natürlichen Zahlen

$$0, 1, 2, 3, \ldots$$

werden von 0 ausgehend durch sukzessive „Nachfolgerbildung" (d.h. Weiterzählen um 1) gebildet. Daher ist die natürlichste Definition einer zahlentheoretischen Funktion (einer für natürliche Zahlen definierten Funktion) die sogenannte *primitive Rekursion*, welche den für 0 angenommenen Funktionswert angibt, ferner die Art, auf welcher aus einem Funktionswert der Funktionswert an der folgenden Stelle berechnet werden kann. Durch endlichmalige Anwendung dieser einfachen Definitionsart nebst Substitutionen kann man von 0 und von der Nachfolgerfunktion ausgehend auch die verwickelten gebräuchlichen Funktionen der elementaren Zahlentheorie erhalten (z.B. die n-te Primzahl als Funktion von n). Auch komplizierter aussehende Definitionsarten der Zahlentheorie, wie z.B. die *simultane rekursive Definition* mehrerer Funktionen, oder die *Wertverlaufsrekursion*, wobei zur Definition des Funktionswertes an einer Stelle beliebig viele Werte des früheren Wertverlaufs verwendet werden können, lassen sich auf primitive Rekursionen und Substitutionen auflösen. Es geben aber auch solche Ausdehnungen des Rekursionsbegriffes (z.B. wobei die Rekursion zugleich nach mehreren Variablen verläuft), die bereits hinausführen von der Klasse der primitiv-rekursiven Funktionen. All diese sind Spezialfälle des von KLEENE eingeführten Begriffes der *allgemeinen Rekursion.*[1]

2. Auf verschiedenen Gebieten der Mathematik hat man mit Mengen zu tun, die ähnlich wie die natürlichen Zahlen aufgebaut werden können: von gewissen „Ausgangselementen" ausgehend (diese spielen die Rolle der 0), durch endlichmalige Anwendung gewisser Funktionen (welche die Rolle der Nachfolgerfunktion spielen). Auch auf solchen „zahlenartig aufbaubaren Mengen" bieten sich als die natürlichsten Definitionsarten für Funktionen die (sinngemäss

[1] Siehe: PÉTER, R., *Rekursive Funktionen.* (Budapest, 1957), 2-te Auflage.

übertragenen) Rekursionen, und für diese gelten ähnliche Sätze wie in der Zahlentheorie.[2]

3. Z.B. entsteht formal betrachtet ein Wort aus dem (üblich durch Λ bezeichneten) ,,leeren Wort'' durch sukzessive Anknüpfung je eines Buchstaben (hier ist belanglos, dass so meistens sinnlose Worte entstehen). Wird eine Menge beliebiger Zeichen als ,,*Alphabet*'' angegeben, so enthält die darauf beruhende ,,*Wortemenge*'' als einziges Ausgangselement Λ, und als Nachfolgerfunktionen dienen die Anknüpfungen der einzelnen Buchstaben des gegebenen Alphabets. Die Wortemengen können vielfach angewandt werden. Ich habe z.B. bewiesen[3], dass die Begriffe der für Programmierung der Rechenautomaten konstruierten Sprache ,,Algol 60'' auf einer geeigneten Wortemenge primitiv-rekursiv sind. Das bezügliche ,,Alphabet'' enthält ausser Buchstaben, Ziffern, Zeichen für logische Werte, Operationszeichen, Klammern und Trennzeichen auch einige fett gedruckten Worte und sogar Wortefolgen, die als einzige ,,Buchstaben'' des ,,Alphabets'' zu betrachten sind (z.B. **else** oder **go to**).

Da die grammatischen Gebilde (z.B. Sätze), die in dieser Arbeit betrachtet werden, durch Nacheinandersetzung von Worten entstehen, so werden sämtliche ,,Buchstaben'' des hier verwendeten ,,Alphabets'' eigentlich Worte sein: das ,,Alphabet'' wird eigentlich ein Vokabular sein. (Freilich können die einzelnen Worte des Vokabulars durch je einen Buchstaben, z.B. durch $v_1, v_2, v_3, \ldots$ bezeichnet werden.)

4. Es ist eine neue Tendenz in der Grammatik, die Eigenschaft ,,grammatisch richtiger Satz zu sein'' genau so exakt zu definieren, wie das in der Mathematik für ,,wohlgebildete Formeln'' geschieht. In vorliegender Arbeit beweise ich, auf Anregung von L. KALMÁR, dem ich die klare Formulierung der mathematischen Grammatiken verdanke, dass diese Eigenschaft sowohl in der ,,kategorialen Grammatik''[4] als auch in der ,,Satzstruktur-Grammatik''[5] in der entsprechenden Wortemenge primitiv-rekursiv ist, in dem Sinne, dass es eine primitiv-rekursive Funktion gibt, die für grammatisch richtige Sätze[6] und nur für solche den Wert Λ annimmt; ferner, dass die grammatisch richtigen Sätze der CHOMSKY-schen ,,Transformations-Grammatik'' (neben sinngemässen Bedingungen) in der entsprechenden Wortemenge rekursiv abzählbar sind, in dem Sinne, dass es eine rekursive Funktion gibt, welche als Werte sämtliche grammatisch richtige Sätze und nur solche annimmt.

Der Gedankengang des § 2 ergibt auch allgemein, dass die Hilfsbegriffe einer beliebigen ,,Metaalgol-Sprache'', falls sie überhaupt definiert sind, und-

[2] PÉTER, R., ,,Über die Verallgemeinerung der Theorie der rekursiven Funktionen für abstrakte Mengen geeigneter Struktur als Definitionsbereiche'', *Acta Math. Acad. Sci. Hungaricae*, I. Teil **12** (1961) S. 271—314; II. Teil **13** (1962) S. 1—24; Berichtigungen dazu in: PÉTER, R., ,,Über die Primitiv-Rekursivität einiger den Aufbau von Formeln charakterisierenden Wortfunktionen'', *Ebenda* **14** (1963) S. 149—172.

[3] PÉTER, R., ,,Primitiv-rekursive Wortbeziehungen in der Programmierungssprache 'Algol 60''', *Publications of the Math. Inst. of the Hungarian Acad. of Sciences* **6** (1961) S. 137—144.

[4] Siehe z.B.: BAR-HILLEL, Y., GAIFMAN, C., SHAMIR, E.: ,,On categorial and phrase-structure grammars:'' *Bulletin of the Research Council of Israel* **9 F** (1960) S. 1—16.

[5] Siehe z. B.: CHOMSKY, N., *Syntactic Structures* ('S-Gravenhage, 1957).

[6] Z.B. ist auch ein Satz wie: ,,Die schlechtgelaunte Gleichung duftet'' — grammatisch richtig.

zwar derart zirkelfrei, dass sie sich nicht selber generieren, primitiv-rekursiv sind.

5. Da sich die folgenden Betrachtungen in Wortemengen vollziehen, zähle ich hier die zur Verwendung kommenden Ergebnisse bezüglich einer Wortemenge M_A über ein Alphabet A der in[2] zitierten Arbeiten auf:

1) Als Vorgänger eines „Wortes" gelten seine zusammenhängende Bestandteile, und für diese wird eine zweckmässige Reihenfolge fixiert. Z.B. sind in der Wortemenge M_A für $a_1, a_2, a_3 \in A$ die Vorgänger des „Wortes" $x = a_1a_2a_3$ in dieser Reihefolge:

$$\Lambda,\ a_1,\ a_2,\ a_1a_2,\ a_3,\ a_2a_3,\ a_1a_2a_3 = x\,.$$

Die von x verschiedenen Vorgänger sind „echte Vorgänger" von x und unter diesen gelten a_1a_2 und a_2a_3, welche nicht echte Vorgänger echter Vorgänger von x sind, als seine „unmittelbaren Vorgänger"; diese enthalten insgesamt als Vorgänger sämtliche echte Vorgänger von x. Die Beziehung „y ist ein Vorgänger (bzw. echter Vorgänger) von x" wird durch

$$y \preceq x \qquad\qquad (\text{bzw. } y \prec x)$$

bezeichnet.

2) In der Wortemenge M_A lautet die primitiv-rekursive Definition einer Funktion $f(x)$:

$$(\mathrm{D})\quad \begin{cases} f(\Lambda) = k\,, \\ \text{für beliebiges } a \in A \\ f(a) = k_a \\ \text{und für beliebige } a, b \in A \\ f(axb) = g_b(ax, xb, f(ax), f(xb))\,, \end{cases}$$

wo k und k_a für jedes $a \in A$ konstante Elemente von M_A und die g_b für jedes $b \in A$ bereits definierte Funktionen sind. (ax und xb sind die unmittelbaren Vorgänger von axb.) Dabei können auch Parameter auftreten: Variablen, die bei der Rekursion unverändert bleiben.

Eine Wortefunktion in M_A ist primitiv-rekursiv, wenn sie von dem leeren Wort Λ und von den Anknüpfungsfunktionen xa (wo $a \in A$) ausgehend durch endlich viele primitive Rekursionen und Substitutionen aufgebaut werden kann.

Eine Wortebeziehung $B(x_1, \ldots, x_n)$ in M_A ist primitiv-rekursiv, wenn ihre charakteristische Funktion primitiv-rekursiv ist, welche etwa wie folgt definiert werden kann:

$$b(x_1, \ldots, x_n) = \begin{cases} \Lambda, \text{ falls } B(x_1, \ldots, x_n) \\ a_0 \text{ sonst}, \end{cases}$$

wobei a_0 ein festgewähltes Element des Alphabets A ist.

3) Die natürlichen Zahlen

$$0,\ 1,\ 2,\ 3, \ldots$$

können in M_A etwa durch

$$\Lambda,\ a_0,\ a_0a_0,\ a_0a_0a_0,\ \ldots$$

vertreten werden; die Bezeichnung a_0^i ist eine kürzere Schreibweise für eine i-gliedrige Kette $a_0 a_0 \ldots a_0$.

4) Die „Ordnung" $o(x)$ eines „Wortes" x bedeutet die Anzahl der Buchstaben von x, und ist eine in M_A primitiv-rekursive Funktion von x.

5) Die „Aneinanderknüpfung" xy ist in M_A eine primitiv-rekursive Funktion von x un y.

6) Die Beziehungen

$$x = y,\ x \prec y,\ x \preceq y$$

sind primitiv-rekursiv in M_A, ebenso zusammen mit B_1 und B_2 auch ihre Disjunktion und Konjunktion

$$B_1 \vee B_2 \text{ und } B_1 \,\&\, B_2 .$$

7) Ist $f(x)$ eine primitiv-rekursive Funktion und $B(x, y)$ eine primitiv-rekursive Beziehung in M_A, dann ist da auch die auf $y \preceq f(x)$ beschränkte „es gibt-Beziehung"

$$(Ey)\, [y \preceq f(x) \,\&\, B(x, y)]$$

primitiv-rekursiv (auch hier können beliebig viele Parameter auftreten).

8) Falls in M_A für jede Wahl von $x_1, x_2, \ldots, x_r$ eine und nur eine der primitiv-rekursiven Beziehungen $B_1(x_1, \ldots, x_r), \ldots, B_n(x_1, \ldots, x_r)$ gilt, so ist auch die aus den primitiv-rekursiven Funktionen $f_1(x_1, \ldots, x_r), \ldots, f_n(x_1, \ldots, x_r)$ „zusammengeflickte" Funktion

$$f(x_1, \ldots, x_r) = \begin{cases} f_1(x_1, \ldots, x_r), \text{ falls } B_1(x_1, \ldots, x_r) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ f_n(x_1, \ldots, x_r), \text{ falls } B_n(x_1, \ldots, x_r) \end{cases}$$

primitiv-rekursiv.

Die letzte Zeile dieser Definition kann auch als

$$„f_n(x_1, \ldots, x_r) \quad \text{sonst"}$$

formuliert werden.

9) Jeder Wortefolge $x_0, x_1, \ldots, x_n$ kann ein „Wort" $c_n(x_0, x_1, \ldots, x_n)$ als für festes n in M_A primitiv-rekursive Funktion derart zugeordnet werden, dass sich daraus die Glieder $x_0, x_1, \ldots, x_n$ der Folge mit Hilfe einer in M_A primitiv-rekursiven Funktion $k_i(x)$ folgendermassen ergeben: für $i = 0, 1, \ldots, n$ ist

$$k_i(c_n(x_0, \ldots, x_n)) = x_i .$$

Dabei gilt für $x \neq \Lambda$ immer

$$k_i(x) \prec x .$$

10) Mit diesen Funktionen lässt sich eine „Wertverlaufsfunktion" $\bar{f}(x)$ der Funktion $f(x)$ folgenderweise angeben: sind

$$\Lambda = \bar{x}_0, \bar{x}_1, \ldots, \bar{x}_m = x .$$

sämtliche Vorgänger von x, in der in 1) angegebenen Reihenfolge, dann sei

$$\bar{f}(x) = c_m(f(\bar{x}_0),\, f(\bar{x}_1), \ldots, f(\bar{x}_m)) .$$

Man sieht, dass $\bar{f}(x)$ die Werte von $f(x)$ für alle Vorgänger der Stelle x zusammenfasst.

Wird in der Definition (D) $f(ax)$ und $f(xb)$ durch $\bar{f}(ax)$ bzw. $\bar{f}(xb)$ ersetzt, so wird aus (D) eine Wertverlaufsrekursion. Weder die Wertverlaufsrekursion, noch die simultane Rekursion mehrerer Funktionen führt von der Klasse der in M_A primitiv-rekursiven Funktionen heraus.

§ 1. Die kategoriale Grammatik

6. Hier hält man vor Augen ein (natürlich endliches) Vokabular, das sämtliche Worte (genauer Wortformen, d.h. z.B. nicht nur ,,Haus'', sondern auch ,,Hauses'', ,,Häuser'', usw.) einer Sprache enthält; wobei neben jedem Wort endlich viele ,,einfache Kategorien'' aufgezählt werden, worunter dasjenige Wort fällt. Diese einfache Kategorien können nicht nur ,,primitive Kategorien'' (wie ,,Hauptwort'', ,,Mehrzahl'', usw.) sein, sondern allgemein entstehen sie aus den primitiven Kategorien, die etwa in einem Anhang zum Vokabular aufgezählt werden, durch endlichmalige Anwendung zweier Operationen, welche wie folgt bezeichnet werden:

$$(x/y) \text{ bzw. } (y\backslash x)\,,$$

da sie sich ähnlich wie die links- bzw. rechtsseitige Division in der Algebra verhalten: bei Nacheinandersetzung von einfachen Kategorien (als Multiplikation) muss nämlich nach Definition

$$(x/y)\,y = x \text{ und } y(y\backslash x) = x$$

gelten. Nämlich einer Kette (Nacheinandersetzung) $w_1 w_2 \ldots w_n$ von Worten lässt man Kategorien, d.h. Ketten (auch eingliedrige) von einfachen Kategorien zuordnen (und zwar erstens sämtliche Ketten je einer der einfachen Kategorien, die der Reihe nach zu $w_1, w_2, \ldots, w_n$ gehören); und dabei werden (x/y) bzw. $(y\backslash x)$ als Kategorien solcher Worteketten erklärt, die vor bzw. nach eine Wortekette der Kategorie y gesetzt, eine Wortekette der Kategorie x erzeugen. Wird z.B. (nach Muster der Kategorien ,,Ausdruck'' und ,,Formel'' in den mathematischen Formelsprachen) eine Kategorie solcher Worteketten, die einen grammatisch richtigen Satz (kurz: ,,Satz'') bilden, mit s bezeichnet, und solcher Worteketten, die in einem Satz Eigennamen vertreten können, ohne die Satz-Beschaffenheit zu zerstören, mit n, so ist $(n\backslash s)$ eine Kategorie solcher Worteketten, die nach eine Wortekette der Kategorie n gesetzt einen Satz erzeugen. Z.B. ist ,,Hans arbeitet'' ein Satz, und auch dann wird daraus ein Satz, wenn darin ,,Hans'' durch eine beliebige Wortekette der Kategorie n (z.B. durch ,,der tüchtige Schmied'') ersetzt wird; daher ist ,,arbeitet'' der Kategorie $(n\backslash s)$.

So charakterisieren die Kategorien einer Wortekette die Rolle dieser Wortekette in Satzbildungen; und da diese Rolle verschieden sein kann, kommen einer Wortekette auch mehrere Kategorien zu.

Kommt in einer Kategorie einer Wortekette ein Teil der Form $(x/y)y$ bzw. $y(y\backslash x)$ vor, und wird dieser durch x ersetzt, so sagt man, dass eine ,,Kürzung'' durchgeführt wurde. Mit einer Kategorie werden auch alle daraus durch Kürzungen entstehende Kategorien einer Wortekette zugeordnet.

So erhält man in der kategorialen Grammatik die Eigenschaft „Satz zu sein" durch folgende Definition: Unter den primitiven Kategorien gibt es eine ausgezeichnete s, und eine Wortekette ist ein Satz, wenn unter den ihr zugeordneten Kategorien auch s vorkommt, d.h. wenn eine ihr zugeordnete Kategorie sich zu s kürzen lässt. Und eine Sprache heisst kategorial, wenn bei geeigneter Wahl der primitiven Kategorien (darunter s) und bei geeigneter Zuordnung von aus diesen aufgebauten einfachen Kategorien zu den Wortformen ihres Vokabulars, ihre Sätze mit den eben als Sätze definierten Worteketten übereinstimmen. Man sieht leicht ein, dass die Formelnsprachen der Mathematik kategorial sind; ob es auch unter den natürlichen Sprachen kategoriale gibt, ist noch unbekannt.

7. In exakter Fassung bedeutet eine kategoriale Grammatik ein geordnetes Quadrupel

$$(V, P, s, Z),$$

wobei V („Vokabular") und P („Menge der primitiven Kategorien") je eine beliebige nicht leere endliche Menge, s ein ausgezeichnetes Element von P („Satz"), und Z die Menge von Zuordnungen ist, welche je einem Element von V endlich viele Elemente der durch P mit den Operationen (x/y) und $(y\backslash x)$ generierten Menge E („Menge der einfachen Kategorien") zuordnet.

Die Elemente e von E sind nach Definition entweder Elemente von P, oder sie besitzen eine der Formen $e = (x/y)$, $e = (y\backslash x)$, wobei x und y auch Elemente von E sind; diese sollen die „unmittelbaren Konstituenten" von e genannt werden. Sämtliche Konstituenten von e sind: e selbst, seine unmittelbaren Konstituenten, die unmittelbaren Konstituenten seiner unmittelbaren Konstituenten, usw.; es ist klar, dass jedes Element von E endlich viele Konstituenten besitzt (ein Element von P enthält als einzige Konstituente sich selbst).

Da Z jedem Element der endlichen Menge V endlich viele Elemente von E zuordnet, so kommen dabei insgesamt nur endlich viele Elemente von E zur Verwendung, und diese besitzen insgesamt nur endlich viele Konstituenten; seien diese

$$k_1, k_2, \ldots, k_r.$$

Unter diesen muss unbedingt auch s auftreten, sonst könnten in der betrachteten Sprache überhaupt keine Sätze vorkommen.

Zu den folgenden Untersuchungen wird die Wortemenge M_V über das Alphabet V zugrunde gelegt. Dann sind die (Wortformen bedeutenden) Elemente des Vokabulars „Buchstaben" des Alphabets V, und die in Nr. 6 behandelten Worteketten sind „Worte" der Wortemenge M_V.

Jedem der „Buchstaben" des Alphabets V wurden durch die Elemente von Z endlich viele E-Elemente zugeordnet. Werden die „Buchstaben" eines Elementes w von M_V durch je einem der ihnen zugeordneten E-Elementen ersetzt, so erhält man eine der zu w gehörigen Kategorien; wird das auf alle mögliche Art vollzogen, und werden auch alle gekürzten Formen der erhaltenen Kategorien hinzugenommen, so ergeben diese insgesamt sämtliche zu w gehörigen Kategorien.

8. Sei v_0 ein beliebiges festgewähltes Element von V. Ferner sei k ein beliebiges der in Nr. 7 eingeführten E-Elementen $k_1, \ldots, k_r$. Unser Ziel ist, in M_V eine Funktion $f_k(x)$ zu definieren, welche den Wert Λ oder v_0 annimmt, je nachdem k unter den zu x gehörigen Kategorien vorkommt oder nicht. Im Fall $k = s$ ist ja $f_k(x)$ die charakteristische Funktion der Eigenschaft „Satz zu sein". Tritt ein $f_{k^*}(x)$ auf, wo k^* nicht unter $k_1, \ldots, k_r$ vorkommt, so soll darunter die Konstante v_0 verstanden werden.

Für das leere Element von M_V kann $f_k(x)$ als v_0 definiert werden.

Seien

$$u_1^{(k)}, u_2^{(k)}, \ldots, u_{n_k}^{(k)}$$

diejenigen Elemente von V, welchen die Kategorie k zugeordnet wurde (diese können aus den endlich vielen Zuordnungen herausgesucht werden). So beginnt die Definition von $f_k(x)$ mit

$$f_k(\Lambda) = v_0$$

und für beliebiges $v \in V$

$$f_k(v) = \begin{cases} \Lambda, \text{ falls } v = u_1^{(k)} \vee v = u_2^{(k)} \vee \ldots \vee v = u_{n_k}^{(k)} \\ v_0 \text{ sonst.} \end{cases}$$

Besteht ein $x \in M_V$ aus mehreren „Buchstaben", so kann ihm die einfache Kategorie k nur so zugeordnet sein, dass man durch Kürzungen einer zu x gehörigen Kategorie zu k gekommen ist, und dabei musste die letzte Kürzung darin bestehen, dass entweder für ein $(k/k_i)k_i$ oder für ein $k_i(k_i\backslash k)$ das k gesetzt wurde. (Da jede Kürzung zu eine Konstituente einer der betroffenen E-Elementen führt, und auch die Konstituenten der Konstituenten eines E-Elementes e Konstituenten von e sind, muss k_i und auch (k/k_i) bzw. $(k_i\backslash k)$ unter $k_1, \ldots, k_r$ vorkommen.) So muss x der Form $x = y_1y_2$ sein, wo y_1 der Kategorie (k/k_i) und y_2 der Kategorie k_i, oder y_1 der Kategorie k_i und y_2 der Kategorie $(k_i\backslash k)$ ist. So gilt für $t_1, t_2 \in V$

$$f_k(t_1xt_2) = \begin{cases} \Lambda, \text{ falls } \bigvee_{i=1}^{r} \{(Ey_1)(Ey_2)[y_1 \leqq t_1x \,\&\, y_2 \leqq xt_2 \,\&\, \\ \qquad \& \, t_1xt_2 = y_1y_2 \,\&\, \\ \qquad \& \, ((f_{(k/k_i)}(y_1) = \Lambda \,\&\, f_{k_i}(y_2) = \Lambda) \vee \\ \qquad \vee (f_{k_i}(y_1) = \Lambda \,\&\, f_{(k_i\backslash k)}(y_2) = \Lambda))]\} \\ v_0 \text{ sonst.} \end{cases}$$

Wird in den Definitionen von $f_k(\Lambda)$, $f_k(v)$ und $f_k(t_1xt_2)$ der Reihe nach $k_1, k_2, \ldots, k_r$ für k gesetzt, so erhält man, wie man leicht sieht, eine simultane Wertverlaufsrekursion für die Funktionen

$$f_{k_1}(x), f_{k_2}(x), \ldots, f_{k_r}(x).$$

Diese kann genau so, wie ich dies an einem Beispiel in meiner in Fusstone[3] zitierten Arbeit ausführlich durchgeführt habe, auf primitiv-rekursive Definitionen der einzelnen Funktionen aufgelöst werden. Da nun unter diesen Funktionen auch $f_s(x)$ vorkommt, und diese die charakteristische Funktion der Eigenschaft „*Satz zu sein*" ist, so ist diese Eigenschaft *in einer kategorialen Grammatik tatsächlich primitiv-rekursiv.*

9. Es ist klar, dass alles genau so geht, wenn nicht nur ein einziges s, sondern beliebig viele (natürlich endlich viele) $s_1, \ldots, s_m$ ausgezeichnet werden (diese können z.B. verschiedene Satzarten bedeuten), und auch dann, wenn unter diesen nicht nur primitive, sondern auch beliebige einfache Kategorien vorkommen.

Kommen darunter auch nicht-einfache Kategorien vor, so muss jede von diesen der Form

$$s = k_{i_1} k_{i_2} \ldots k_{i_e}$$

sein, wo $i_1, i_2, \ldots, i_e$ unter $1, 2, \ldots, r$ vorkommende Zahlen sind, und die Definitionsgleichungen

$$f_s(t_1 x t_2) = \begin{cases} \Lambda, \text{ falls } (Ey_1) \ldots (Ey_c)\,[y_1 \leqq t_1 x \mathbin{\&} \ldots \mathbin{\&} y_{c-1} \leqq t_1 x \mathbin{\&} \\ \quad \mathbin{\&} y_c \leqq x t_2 \mathbin{\&} t_1 x t_2 = y_1 y_2 \ldots y_c \mathbin{\&} \\ \quad \mathbin{\&} f_{k_{i_1}}(y_1) = \Lambda \mathbin{\&} \ldots \mathbin{\&} f_{k_{i_c}}(y_c) = \Lambda] \\ v_0 \text{ sonst} \end{cases}$$

können zu den bisherigen hinzugenommen werden, ohne die simultane Wertverlaufsrekursionsbeschaffenheit der Definition zu stören.

§ 2. Die Satzstruktur-Grammatik

10. Hier handelt es sich um ähnliche Definitionen wie im „Metaalgol", mit der Abweichung, dass hier das Zeichen :: = als „ist nach einer der möglichen Definitionen" zu lesen ist. Z.B. lautet eine solche Definition des arithmetischen Ausdrucks (wobei, wie üblich, x' den Nachfolger von x bezeichnet, und ⟨Ausdruck⟩ keinen konkreten, sondern einen beliebigen Ausdruck bedeutet; ähnliches gilt auch für die anderen Anwendungen solcher spitzigen Klammern; is ist belanglos, dass CHOMSKY eine andere Bezeichnung statt spitzigen Klammern verwendet, und auch die Spazien zwischen den Worten irgendwie bezeichnet hat):

⟨Ausdruck⟩ : : = 0
⟨Ausdruck⟩ : : = ⟨Variable⟩
⟨Ausdruck⟩ : : = ⟨Ausdruck⟩'
⟨Ausdruck⟩ : : = (⟨Ausdruck⟩ ⟨Zeichen einer zweigliedrigen Operation⟩ ⟨Ausdruck⟩)
⟨Zeichen einer zweigliedrigen Operation⟩ : : = +
⟨Zeichen einer zweigliedrigen Operation⟩ : : = .
⟨Variable⟩ : : = ⟨Buchstabe⟩
⟨Variable⟩ : : = ⟨Buchstabe⟩ ⟨Index⟩
⟨Buchstabe⟩: : = a
⟨Buchstabe⟩: : = b
. . . .
⟨Buchstabe⟩: : = z
⟨Index⟩ : : = ⟨von 0 verschiedene untere Ziffer⟩
⟨Index⟩ : : = ⟨Index⟩ ⟨untere Ziffer⟩
⟨untere Ziffer⟩ : : = $_0$
⟨untere Ziffer⟩ : : = ⟨von 0 verschiedene untere Ziffer⟩
⟨von 0 verschiedene untere Ziffer⟩ : : = $_1$
. . . .
⟨von 0 verschiedene untere Ziffer⟩ : : = $_9$

Chomsky versucht ähnlich den „Satz"-Begriff der natürlichen Sprachen durch Abbau seiner Struktur bis auf konkrete Worte eines Vokabulars zu definieren.

11. So ist in exakter Fassung eine Satzstruktur-Grammatik ein geordnetes Quadrupel

$$(T, H, s, P),$$

wo T (das „terminale Vokabular" für Worte, die für sich stehen) und H („Hilfsvokabular" für die Hilfsbegriffe in spitzigen Klammern) beliebige endliche, nicht leere Mengen sind, und s ($\langle$Satz$\rangle$) ein ausgezeichnetes Element von H ist. Ferner ist P eine endliche Menge von Definitionsgleichungen im Sinne der „möglichen Gleichheit $::=$", die Produktionen genannt werden. Genau formuliert hat eine Produktion die Form

$$w ::= w_1 w_2 \ldots w_n,$$

wo $w \in H$ und $w_1, w_2, \ldots, w_n \in T + H$.

Jeder Hilfsbegriff muss definiert werden, daher kommt jedes Element von H auf der linken Seite von $::=$ mindestens eines Elementes von P vor. Seien sämtliche Hilfsbegriffe

$$h_1, h_2, \ldots, h_l.$$

Sind für ein $t \in T$ die Produktionen

$$h_{i_1} ::= t$$

$$h_{i_2} ::= t$$

$$\cdots\cdots$$

$$h_{i_r} ::= t$$

Elemente von P, dann sollen zu t im Vokabular T diese Kategorien $k_{i_1}, k_{i_2}, \ldots, k_{i_r}$ zugeordnet werden.

Eine Kette $w_1 w_2 \ldots w_n$ mit $w_1, w_2, \ldots, w_n \in T + H$ soll eine „Konstruktion" genannt werden, und n die Ordnung dieser Konstruktion.

Man sagt, eine Konstruktion ψ wird durch die Konstruktion $\varphi = v_1 \ldots v_m$ „unmittelbar generiert", falls für ein $1 \leqq i \leqq m$

$$\psi = v_1 \ldots v_{i-1} w_1 \ldots w_n v_{i+1} \ldots v_m$$

ist, wobei

$$v_i ::= w_1 \ldots w_n$$

unter den Elementen von P vorkommt. Und eine Konstruktion ψ wird durch eine Konstruktion φ „generiert", wenn es eine Folge

$$(*) \qquad \varphi_1 = \varphi, \varphi_2, \ldots, \varphi_r = \psi$$

von Konstruktionen gibt, wobei φ_i für $i = 2, 3, \ldots, r$ durch φ_{i-1} unmittelbar generiert wird.

Eine Konstruktion $w_1 w_2 \ldots w_n$ heisst „terminal", wenn $w_1, w_2, \ldots, w_n \in T$, also wenn darin schon keine Hilfsbegriffe vorkommen.

Nun wird in einer Satzstruktur-Grammatik eine Konstruktion „Satz” genannt, falls sie terminal ist, und durch s generiert wird. Auch das ist bis heute nicht geklärt, ob es zu einer natürlichen Sprache eine Satzstruktur-Grammatik gibt, deren Sätze mit den Sätzen der betrachteten natürlichen Sprache übereinstimmen.

12. Damit die Definition des „Satz”-es nicht zirkelhaft ausfallen soll, muss die Bedingung gestellt werden, dass kein Element von H sich selbst generieren soll.

Da nach Definition die Ordnung einer generierten Konstruktion nie kleiner als die Ordnung der generierenden Konstruktion sein kann, könnte ein $h_{i_1} \in H$ sich selber nur durch eine Folge aus Elementen von P

$$(**) \qquad h_{i_1} ::= h_{i_2}, \quad h_{i_2} ::= h_{i_3}, \ldots, h_{i_r} ::= h_{i_{r+1}}$$

generieren, wo $h_{i_{r+1}} = h_{i_1}$ ist. (Natürlich kann hier auch $r = 1$, also eine einzige Produktion $h_{i_1} ::= h_{i_1}$ stehen.) Es muss also verlangt werden, dass keine solche Elementenfolge in P vorkommen soll. Auch für $h_{i_{r+1}} \neq h_{i_1}$ darf keine solche Elementenfolge in P existieren, falls $r \geqq l$ ist, weil dabei unter $r + 1$ Indizes $i_1, \ldots, i_{r+1}$ unbedingt auch gleiche auftreten müssen, und z.B. für $i_u = i_v$ mit $1 \leqq u < v \leqq r + 1$ würde mittels eines Teils der Produktionenfolge (**) h_{i_u} durch sich selber generiert.

Nach dieser Bedingung kann eine generierende Konstruktionenfolge der Art (*) in Nr. 11 nur für $j \leqq l$ mit

$$\varphi_1 = h_{i_1}, \quad \varphi_2 = h_{i_2}, \ldots, \varphi_j = h_{i_j}$$

beginnen; fortgesetzt werden kann es dann entweder durch ein Element von T, oder durch eine Konstruktion mindestens 2-ter Ordnung. Und unbedingt kann es fortgesesetzt werden, da die Forderung gestellt wurde, dass jedes $h_i \in H$ an der linken Seite von $::=$ mindestens eines Elementen von P vorkommen muss.

13. Die in P auftretenden Produktionen geben nur je eine mögliche Definition je eines Hilfsbegriffes h_i. Die vollständige Definition eines h_i ist eine Alternative: sind die rechten Seiten von $::=$ in jenen Produktionen, in welchen auf der linken Seite h_i steht, die (eventuell vorkommenden) terminalen Konstruktionen

$$\omega_1^{(i)}, \ldots, \omega_{r^{(i)}}^{(i)}$$

und nicht-terminalen Konstruktionen

$$\varphi_1^{(i)} = \omega_{1,1}^{(i)} h_{1,1}^{(i)} \omega_{1,2}^{(i)} h_{1,2}^{(i)} \ldots \omega_{1,r_1^{(i)}}^{(i)} h_{1,r_1^{(i)}}^{(i)} \omega_{1,r_1^{(i)}+1}^{(i)}, \ldots$$

$$\ldots, \varphi_{j_i}^{(i)} = \omega_{j_i,1}^{(i)} h_{j_i,1}^{(i)} \ldots h_{j_i,r_{j_i}^{(i)}}^{(i)} \omega_{j_i,r_{j_i}^{(i)}+1}^{(i)},$$

wo jedes h mit beliebigen Indizes eines der $h_1, \ldots, h_l$ ist, und jedes ω mit beliebigen Indizes entweder leer oder eine terminale Konstruktion ist, so ist

$$h_i = \omega_1^{(i)} \text{ oder } \ldots \text{ oder } \omega_{r^{(i)}}^{(i)} \text{ oder } \varphi_1^{(i)} \text{ oder } \ldots \text{ oder } \varphi_{j_i}^{(i)}.$$

Bestehen hier einige der $\varphi_1^{(i)}, \varphi_2^{(i)}, \ldots, \varphi_{j_i}^{(i)}$ aus je einem einzigen h-Element von H, so können diese durch die diese Elemente definierende Alternativen ersetzt werden, und so wird h_i durch eine eventuell mehrgliedrige Alternative definiert. Darauf kann dasselbe wiederholt werden, usw., aber nur in höchstens

l Schritten, da sonst eine in Nr. 12 ausgeschlossene generierende Konstruktionenfolge vorhanden wäre. So kann man annehmen, dass bereits in der obigen vollständigen Definition von h_i jedes $\varphi_u^{(i)}$ eine Konstruktion mindestens 2-ter Ordnung ist, so dass jedes darin vorkommende $h \in H$ ein echter Bestandteil von ihm ist.

14. Nun kommt es darauf an, die charakteristische Funktion $g_i(x)$ der Eigenschaft „ein Hilfsbegriff h_i zu sein" d.h. „durch h_i generierbar zu sein" für $i = 1, 2 \ldots, l$ in der Wortemenge M_T über das Alphabet T als primitiv-rekursiv zu definieren, da unter diesen Hilfsbegriffen auch das ausgezeichnete s („Satz") vorkommt.

Sei auch hier

$$g_i(\Lambda) = t_0,$$

wo t_0 ein festgewähltes Element von T bezeichnen soll.

Seien ferner

$$t_1^{(i)}, t_2^{(i)}, \ldots, t_n^{(i)}$$

diejenigen Elemente von T, welchen nach Nr. 11 die Kategorie h_i zugeordnet wurde. Dann gilt für ein beliebiges $t \in T$

$$g_i(t) = \begin{cases} \Lambda, \text{ falls } t = t_1^{(i)} \vee t = t_2^{(i)} \vee \ldots \vee t = t_n^{(i)} \\ t_0 \text{ sonst.} \end{cases}$$

Wenn endlich die in Nr. 13 angegebene vollständige Definition von h_i in Betracht gezogen, und die charakteristische Funktion von „ein Hilfsbegriff $h_{c,u}^{(i)}$ zu sein" für $c = 1, 2, \ldots, j_i$; $u = 1, 2, \ldots, r_c^{(i)}$ mit $g_{c,u}^{(i)}$ bezeichnet wird, so ist für beliebige $t_1, t_2 \in T$

$$g_i(t_1xt_2) = \begin{cases} \Lambda, \text{ falls } t_1xt_2 = \omega_1^{(i)} \vee \ldots \vee t_1xt_2 = \omega_{r^{(i)}}^{(i)} \vee \\ \vee \bigvee_{c=1}^{j_i} \{(Ey_1)(Ey_2) \ldots (Ey_{r_c^{(i)}}) [y_1 \leqq t_1x \mathbin{\&} \\ \mathbin{\&} y_2 \leqq t_1x \mathbin{\&} \ldots \mathbin{\&} y_{r_c^{(i)}} \leqq xt_2 \mathbin{\&} \\ \mathbin{\&} t_1xt_2 = \omega_{c,1}^{(i)} y_1 \omega_{c,2}^{(i)} y_2 \ldots y_{r_c^{(i)}} \omega_{c,r_c^{(i)}+1}^{(i)} \mathbin{\&} \\ \mathbin{\&} g_{c,1}^{(i)}(y_1) = \Lambda \mathbin{\&} \ldots \mathbin{\&} g_{c,r_c^{(i)}}^{(i)}(y_{r_c^{(i)}}) = \Lambda]\} \\ t_0 \text{ sonst,} \end{cases}$$

wobei die ω mit allen vorkommenden Indizes feste Elemente von M_T sind (und freilich nicht alle Disjunktionsglieder auftreten müssen).

Genau so, wie zum Schluss der Nr. 8 geschildert wurde, sieht man ein, dass die Definitionen von $g_i(\Lambda)$, $g_i(t)$ und $g_i(t_1xt_2)$ für $i = 1, 2, \ldots, l$ eine simultane Wertverlaufsrekursion für die Funktionen $g_i(x)$ liefern, woraus sich diese Funktionen als primitiv-rekursiv in M_T ergeben. Da unter diesen Funktionen auch die charakteristische Funktion der Eigenschaft *„Satz zu sein"* vorkommt, so ist diese Eigenschaft *in einer Satzstruktur-Grammatik primitiv-rekursiv*.

Der Gedankengang war ganz allgemein, so ergibt sich daraus, dass *in einer beliebigen Metaalgol-Sprache* — die aus einem Tripel (T, H, P) mit T, H, P in unserem Quadrupel (T, H, s, P) beschriebener Art besteht, wo belanglos ist, ob gewisse Hilfsbegriffe ausgezeichnet werden — *sämtliche Hilfsbegriffe,* falls es überhaupt je eine definierende Produktion für sie vorliegt, und sie sich nicht selber generieren können, in der entsprechenden Wortemenge *primitiv-rekursiv* sind.

15. Die Vorangehenden beziehen sich auf die „Kontextunabhängige" Satzstruktur-Grammatik. CHOMSKY unterscheidet auch eine „kontextbedingte" Satzstruktur-Grammatik, worin die Produktionen je einen Hilfsbegriff in verschiedenen Kontexten verschieden definieren. Hier hat also eine Produktion die Form:

$$\mu h_i \nu ::= \mu\omega\nu\,,$$

wo $h_i \in H$, ferner μ, ν und ω Konstruktionen sind. Kommen z.B. Produktionen

⟨Satz⟩ : : = ⟨Subjektteil⟩ ⟨Prädikatteil⟩
⟨Subjektteil⟩ : : = der Schmied
⟨Subjektteil⟩ : : = die Arbeiter
⟨Prädikatteil⟩ : : = arbeitet
⟨Prädikatteil⟩ : : = essen

vor, so können daraus die grammatisch falschen Sätze „der Schmied essen" und „die Arbeiter arbeitet" generiert werden. Das Prädikat „arbeitet" dürfte nur nach Subjekte wie „der Schmied", das Subjekt „die Arbeiter" nur vor Prädikate wie „essen" gesetzt werden. Das bedeutet die Verwendung solcher Produktionen:

Der Schmied ⟨Prädikatteil⟩ : : = Der Schmied arbeitet
⟨Subjektteil⟩ essen : : = die Arbeiter essen.

Das kann aber dadurch erledigt werden, dass man als Hilfsbegriffe auch „Subjektteil in Einzahl", „Subjektteil in Mehrzahl", „Prädikatteil in Einzahl", „Prädikatteil in Mehrzahl" einführt, und folgende Produktionen anwendet:

⟨Satz⟩ : : = ⟨Subjektteil in Einzahl⟩ ⟨Prädikatteil in Einzahl⟩
⟨Satz⟩ : : = ⟨Subjektteil in Mehrzahl⟩ ⟨Prädikatteil in Mehrzahl⟩.

Ähnlich können auch andere kontextbedingte Fälle erledigt werden, und so vermutlich jede praktisch auftretende kontextbedingte Satzstruktur-Grammatik auf kontextunabhängige Satzstruktur-Grammatik zurückgeführt werden.

16. CHOMSKY hat einen Satz eigentlich nicht als Wortekette, sondern als Morphemenkette betrachtet, wo z.B. „Haus" eine Stamm-Morpheme ist, aus welcher „Häuser" durch Anwendung der Zusatz-Morphemen der Umlautbildung und des Anknüpfen der Mehrzahl-Endung (welche vom Stammwort abhängig manchmal nichts, sonst -e, -n, -er oder -en ist) gebildet wird. In exakter Fassung sind also die Zusatz-Morpheme Funktionen (seien die „einfachen" darunter mit $z_1, z_2, \ldots, z_m$ bezeichnet), die für Stamm-Morpheme erklärt sind und als Werte Elemente von T annehmen. Ketten einfacher Zusatz-Morphemen-Zeichen ergeben die Zeichen der notwendigen zusammengesetzten Funktionen, z.B. einer Funktion, die auf die Stamm-Morpheme eines Zeitwortes angewandt, dieses zugleich in Mehrzahl, erste Person und Imperfekt setzt. Der Wert einer Funktion $z_{i_1} \ldots z_{i_r}$ an der Stelle t soll mit $z_{i_1} \ldots z_{i_r} t$ bezeichnet werden. Unter $z_1, \ldots, z_m$ muss auch das Zeichen der nichts ändernden Identitätsfunktion $z_i x = x$ vorkommen.

In dieser Auffassung ersetzt man das aus Wortformen bestehende terminale Vokabular T einer Satzstruktur-Grammatik durch ein Morphemen-

Vokabular T^*, welches alle Stamm-Morpheme und die Funktionszeichen $z_1, \ldots, z_m$ enthält. Alles andere geht dann genau so wie in der erst betrachteten Satzstruktur-Grammatik, darum beschränken sich die Weiteren auf eine solche.

§ 3. Transformations-Grammatik

17. Am nähesten zur Generierung der Sätze der natürlichen Sprachen scheint die Transformations-Grammatik zu führen. Hier handelt es sich darum, dass nur die einfachsten Sätze, die den ,,Kern" der Sprache ausmachen, die sogenannten ,,Kernsätze" durch eine Satzstruktur-Grammatik generiert werden, und alle Sätze entstehen aus diesen durch iterierte Anwendung endlich vieler ,,Transformationen"; ähnlich wie die wahren Formeln einer axiomatischen mathematischen Theorie aus den Axiomen durch iterierte Anwendung gewisser Schlussregeln (welche gewisse Formeln in eine neue Formel überführen) generiert werden. Zum Beispiel kann ein Kernsatz *Dieses Haus ist schön* in seine Verneinung: *Dieses Haus ist nicht schön*, oder in eine der verschiedenen Frageformen transformiert werden: *Ist dieses Haus schön? Welches Haus ist schön? Wie schaut dieses Haus aus? Was ist dieses schöne Ding?* — oder die beiden Sätze:

Ein Märchen kommt mir nicht aus dem Sinn
Das Märchen ist aus alten Zeiten

lassen sich durch eine ,,eigenschafteinführende Transformation" in den Satz:

Ein Märchen aus alten Zeiten das kommt mir nicht aus dem Sinn

transformieren.

Die Transformationen, die in einer Sprache verwendet werden, können annehmlich so beschrieben werden, dass sie sich in exakter Fassung in der geeigneten Wortemenge als rekursiv erweisen.

Da eine axiomatische Theorie allgemein nicht rekursiv entscheidbar ist, kann man allgemein nicht erwarten, dass der Satzbegriff einer Transformations-Grammatik primitiv- oder auch nur allgemein-rekursiv sei. Ich zeige aber, dass er rekursiv-abzählbar ist.

18. Sei eine Satzstruktur-Grammatik (T, H, s, P) zum Generieren der Kernsätze gegeben, und M_T die Wortemenge über das Alphabet T. Sämtliche zulässige Transformationen, welche Elemente $x_1, \ldots, x_r$ von M_T in ein Element von M_T überführen, seien die Funktionen

$$f_1(x_1, \ldots x_r), \ldots, f_n(x_1, \ldots, x_r)$$

(dabei kann r eine feste Zahl sein, da die Funktionen auch fiktive Variablen besitzen können, in welchen sie konstant sind). Zu diesen sollen der Reihe nach die Beziehungen

$$B_1(x_1, \ldots, x_r), \ldots, B_n(x_1, \ldots, x_r)$$

gehören, welche bestehen müssen, damit die entsprechenden Transformationen Sätze $x_1, \ldots, x_r$ (wobei jetzt auch der leere Satz zugelassen wird) wieder in Sätze überführen sollen (es würde z.B. die in Nr. 17 erwähnte ,,eigenschafteinführende Transformation" nicht aus beliebigen Sätzen einen Satz erzeugen; es war wesentlich, dass die beiden Sätze des Beispiels dasselbe Subjekt hatten). Sei endlich $s(x)$ die nach § 2 primitiv-rekursive charakteristische Funktion der Eigenschaft ,,Kernsatz zu sein".

Dann kann mit Hilfe der in Nr. 5 eingeführten primitiv-rekursiven Funktion $k_i(x)$ folgenderweise eine Funktion $f(x)$ definiert werden, welche als Werte nur Sätze (darunter auch den „leeren Satz Λ") und sämtliche Sätze annimmt, welche also die Sätze „abzählt":

$$f(\Lambda) = \Lambda$$

und für $x \neq \Lambda \in M_T$

$$f(x) = \begin{cases} k_1(x), \text{ falls } k_0(x) = \Lambda \mathbin{\&} s(k_1(x)) = \Lambda \\ f_1\big(f(k_1(x)), \ldots, f(k_r(x))\big), \text{ falls } o(k_0(x)) = 1 \mathbin{\&} \\ \qquad\qquad \mathbin{\&} B_1\big(f(k_1(x)), \ldots, f(k_r(x))\big) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ f_n\big(f(k_1(x)), \ldots, f(k_r(x))\big), \text{ falls } o(k_0(x)) = n \mathbin{\&} \\ \qquad\qquad \mathbin{\&} B_n\big(f(k_1(x)), \ldots, f(k_r(x))\big) \\ \Lambda \text{ sonst.} \end{cases}$$

Erstens sind nämlich sämtliche Werte von $f(x)$ (eventuell leere) Sätze: für $x \neq \Lambda$, $k_0(x) = \Lambda$ sind sie Kernsätze, da $s(x)$ genau für Kernsätze gleich Λ ist; ferner gilt $f(\Lambda) = \Lambda$ als Satz, und falls $f(\bar{x})$ für jedes $\bar{x} \prec x$ bereits ein Satz ist, so ist auch für $k_0(x) \neq \Lambda$, da

$$k_1(x) \prec x, \ldots, k_r(x) \prec x$$

gilt, nach der Definition auch $f(x)$ ein Satz. So gilt die Behauptung der Reihe nach bei $o(x) = \Lambda$, $o(x) = 1$, $o(x) = 2, \ldots$, also für jedes x.

Zweitens kommen unter den Werten von $f(x)$ sämtliche Sätze (mehrmals) vor. Ist nämlich zunächst y ein Kernsatz, dann gilt nach Nr. 5. unter anderem für

$$x = c_1(\Lambda, y)$$

laut Definition

$$f(x) = y\,.$$

Sind ferner $y_1, \ldots, y_r$ Sätze, für welche $B_i(y_1, \ldots, y_r)$ bei einem der $i = 1, 2, \ldots, n$ gilt, und wurden diese bereits von $f(x)$ angenommen, etwa der Reihe nach an Stellen $x_1, \ldots, x_r$, d.h. gilt

$$f(x_1) = y_1, \ldots, f(x_r) = y_r\,,$$

so ist unter anderem für

$$x = c_r(t_0^i, x_1 \ldots, x_r)$$

laut Definition

$$f(x) = f_i(y_1, \ldots, y_r)\,.$$

Die Sätze der Transformations-Grammatik werden aber alle auf diese Art generiert.

Man sieht ganz ähnlich wie am Beispiel in Nr. 17—18 mit Benutzung von Nr. 30 des I. Teils meiner in Fussnote[2] zitierten ersten Arbeit ein, dass falls die Funktionen $f_1, \ldots, f_n$ und die Beziehungen $B_1, \ldots, B_n$ primitiv- (bzw.

allagemein-) rekursiv sind, so die Definition von $f(x)$ eine Wertverlaufsrekursion, und daher $f(x)$ eine primitiv- (bzw. allgemein-) rekursive Funktion in M_T ist. *Daher ist in diesem Fall der Satzbegriff einer Transformations-Grammatik primitiv- (allgemein-) rekursiv abzählbar.*

(Eingegangen: 24. Juli, 1963.)

О РЕКУРСИВНОСТИ ПОНЯТИЙ МАТЕМАТИЧЕСКИХ ГРАММАТИК

R. PÉTER

Резюме

Наиболее естественным способом определения теоретико-числовых функций, соответственно образованию натуральных чисел по определенным шагам исходя из нуля, является примитивная рекурсия. Известно, что сложные функции, употребляемые в элементарной теории чисел, также являются примитивно-рекурсивными, то есть, исходя из нуля и из последующего образования они могут быть заданы через конечное число примитивных рекурсий и подстановок.

В различных областях математики находят применение множества, построенные из некоторых основных элементов по образцу множества натуральных чисел путем применения некоторых функций конечное число раз. Появляется мысль о соответствующем обобщении рекурсии, как наиболее естественном способе построения функций, определенных на таких множествах; и так по большей части обобщения результатов, полученных в теории чисел, остаются в силе и для обобщенных рекурсивных функций.

С точки зрения приложений особенно важен случай «множества слов», в котором, исходя из пустого слова (его символ Λ), мы строим его элементы последовательными прибавлениями элементов («букв») произвольного заданного множества («альфавита»). В настоящей статье используются такие «альфавиты», которые в каждом случае означают по одному словарю и поэтому его «буквы» являются собственно говоря словами.

Настоящая статья связана именно с тем новым грамматическим стремлениям, согласно которому понятие «грамматически правильной фразы» (коротко «фразы») должна быть определено таким же точным способом, как в математике может быть определено понятие «правильно построенной формулы». «Категоричная грамматика», «форазово-структурная грамматика» и «трансформационная грамматика» — они все три дают различные определения этого понятия; до сих пор еще не решено, существует ли (кроме математических языков-формул) также такой естестенный язык, для которого применение этого понятия действительно дает все фразы данного языка.

В настоящей статье по инициативе L. KALMÁRA автор доказывает, что понятие фразы как в категоричной грамматике, так и в фразово-структурной грамматике является примитивно-рекурсивным на соответствующем множестве слов в том смысле, что существует функция, которая принимает значение Λ, если ее применяют к фразам и только к фразам; далее автор доказывает, что множество фраз трансформационной грамматики является

рекурсивно-перечислимым на соответствующем множестве слов (при разумных условиях) в том смысле, что существует такая рекурсивная функция, которая принимает в качестве значений все фразы и только их.

Общий характер доказательства относящегося к фразово-структурной грамматике, сразу дает в общем виде тот результат, что понятия любого «мэтаалгольного языка» являются примитивно-рекурсивными, если только они вообще появляются вследствии их определения, а именно без ложного круга, в том смысле, что они не смогут «производить» самых себя (в некотором смысле, заданном точным способом).