Acta Informatica 4, 55—75 (1974)

Sequentielle Analyse kontextfreier Sprachen

G. Hotz

Eingegangen am 27. März 1974

Summary. Sequential analysis of context-free languages. Since 1965 several methods to solve the word problem for context-free languages in no more than cn^3 steps have been published where c is a constant and n the length of the word [2, 5, 7]. The procedure to be described is a sequential procedure that means that the word which is to be analyzed has to be read monotonic from left to right and that the word problem is decided for each partial word. The method consists in constructing a "growing" automaton whose wiring (in a physical way of speaking) grows only quadratically. From a monotonicity of property of signal propagation in our automata we can conclude that the procedure can be implemented on a RAM (random access machine) in such a way that the word problem is decided in at most cn^3 steps.

1. Notation und der sequentielle syntaktische Automat

Wir bezeichnen mit T das *Terminalalphabet*, mit Z das *Zwischenalphabet* oder *nichtterminale Alphabet*. In unseren Konstruktionen werden weitere Zwischenalphabete Y und indizierte Alphabete $Y_i^{t,z} \subset Y$ verwendet werden. Z tritt in diesen Fällen als Terminalalphabet auf. Das Axiom der vorgegebenen kontextfreien Sprache wird stets s sein.

Wir nehmen an, daß die gegebene kontextfreie Grammatik in *Chomsky-Normal-Form* vorliegt; das heißt für das Produktionssystem P' unserer Grammatik gilt

$$P' = P \cup Q$$

mit

$$P \subset Z \times (Z - s) \times (Z - s), \qquad Q \subset Z \times T \cup \{s, 1\},$$

wo 1 das leere Wort bezeichnet, und es kommen keine überflüssigen Alphabetzeichen und Produktionen vor.

Die Produktionen von P und Q auf die erste Komponente bezeichnen wir mit D (=domain), die Projektion auf die zweite Komponente mit C (=codomain).

Es ist uns bequem, die Grammatiken als

$$G = (Z \cup T, T, P, Q, s)$$

zu notieren, also für die beiden Produktionstypen verschiedene Bezeichnungen einzuführen.

Wir setzen

$$L(G) = \{w \in T^* \mid s \rightarrow w\},$$

worin $\rightarrow$ die Ableitung nach G bezeichnet.

Weiter ist für $L \subset T^*$

$$F(L) = \{t \in T \mid \text{Es gibt } w \in T^* \text{ mit } t \cdot w \in L\}$$

und

$$L_u = \{w \in T^* \mid u \cdot w \in L\}$$

für

$$u \in T^*.$$

Offensichtlich gilt für $L, L' \subset T^*$.

$$L_1 = L, \qquad L_{u \cdot t} = (L_u)_t,$$

$$(L \cup L')_u = L_u \cup L'_u,$$

$$F(L \cup L') = F(L) \cup F(L')$$

und

$$F\big(L(G)\big) \subset C(Q).$$

Die Relation

$$u \equiv v \Leftrightarrow L_u L_v$$

ist die von Rabin-Scott [6] eingeführte *syntaktische Kongruenz* nach L.

Wir definieren nun den *sequentiellen syntaktischen* Automaten $\mathscr{S}$ mit dem Eingabealphabet $I(\mathscr{S}) = T$, dem *Ausgabealphabet* $O(\mathscr{S}) = \{0, 1\}$ und der *Zustandsmenge* $S(\mathscr{S}) = \wp T^*$, wo $\wp$ die Potenzmenge bezeichnet, und dem Anfangszustand L.

Die *Überführungsfunktion* δ bzw. *Ausgabefunktion* λ ist wie folgt definiert:

$$\delta(t, M) = Mt \quad \text{für} \quad M \subset T^*, \qquad t \in T$$

und

$$\lambda(M) = 1 \Leftrightarrow 1 \in M.$$

Die durch $\mathscr{S}$ akzeptierte Menge $\Lambda(\mathscr{S})$ ist

$$\Lambda(\mathscr{S}) = \{w \in T^* \mid \lambda\big(\delta(w, L)\big) = 1\}.$$

Man zeigt leicht

$$L = \Lambda(\mathscr{S}) \text{ und } \mathscr{S} \text{ is } \textit{reduziert}.$$

Wir bezeichnen mit $\mathscr{S}(L)$ den Teilautomaten, der genau die von L aus erreichbaren Zustände enthält. Bekanntlich gilt

$\mathscr{S}(L)$ ist genau dann endlich, wenn $\mathscr{L}$ regulär ist.

Da es nicht generell entscheidbar ist, ob eine kontextfreie Sprache regulär ist, gilt weiter

„$\mathscr{S}(L)$ endlich" ist nicht generell entscheidbar.

Unser Entscheidungsverfahren ist ein zu $\mathscr{S}(L)$ äquivalenter sequentieller Automat. Wir untersuchen nicht, ob er reduziert ist. Wie aus obigem folgt, ist es nicht entscheidbar, ob er bei gegebenem L in Wirklichkeit ein endlicher Automat ist.

2. Die Z-Basis von G

Wir verwenden nun die Terminologie und Notation der zu G gehörigen freien X-Kategorie [3]. Das heißt, daß wir mit den Klassen *nicht wesentlich verschiedener* Ableitungen der Wörter $w \in (Z \cup T)^*$ rechnen. Diese Klassen bezeichnen wir mit $f, g, h, \ldots$. Ist $w \to v$, dann gibt es mindestens ein f mit

$$w \xrightarrow{f} v.$$

Wir schreiben dann auch

$$w = D(f), \qquad v = C(f).$$

Die Hintereinanderausführung von Ableitungen f und g ist definiert, wenn $D(f) = C(g)$ ist und wird $f \circ g$ bezeichnet. Die Parallelausführung bezeichnen wir mit $f \times g$. Es gilt offensichtlich

$$D(f \circ g) = D(g), \qquad C(f \circ g) = C(f)$$

$$D(f \times g) = D(f) \cdot D(g), \qquad C(f \times g) = C(f) \cdot C(g),$$

$$(f_1 \circ g_1) \times (f_2 \circ g_2) = (f_1 \times f_2) \circ (g_1 \times g_2)$$

für

$$D(f_i) = C(g_i), \qquad i = 1, 2.$$

Wir setzen weiter

$$\mathcal{M}_G(M_1, M_2) = \{f \mid D(f) \in M_1,\ C(f) \in M_2\}.$$

$\mathcal{M}_G(M_1, M_2)$ enthält also die Menge aller Ableitungsklassen von Wörtern aus M_1 in Wörtern aus M_2.

Sei nun für $u \in T^*$ und $z \in Z$

$$\mathcal{B}^u(G, z) = \{f \in \mathcal{M}_G(z, u \cdot Z^*) \mid f\, u\text{-prim}\}.$$

f heißt *u-prim*, wenn aus

$$f = (1_u \times g) \circ h \quad \text{mit} \quad u \cdot w = C(f)$$

folgt $g = 1_w$. 1_w und 1_u sind die Identitäten auf w bzw. u.

Es gilt

Lemma 1. Zu jedem $f \in \mathcal{M}_G(s, u(Z \cup T)^*)$ gibt es genau eine Zerlegung

$$f = (1_u \times g) \circ h$$

mit $h \in \mathcal{M}_G(s, u \cdot Z^*)$ und $h\,u$-prim.

Da wir die Eindeutigkeit hier nicht verwenden und der Rest der Behauptung trivial ist, beweisen wir das Resultat nicht, sondern verweisen auf [4].

Wir schreiben nun

$$C(\mathcal{M}) = \{C(f) \mid f \in \mathcal{M}\}$$

und definieren hiermit

$$B^u(z) = \big(C(\mathcal{B}^u(G, z))\big)_u.$$

$B^u(z)$ heißt *(u, z)-Basis* von G.

Es ist offensichtlich

$$B^u(z) \subset Z^*.$$

Darüber hinaus gilt

Lemma 2.
$$L_u = L(Z \cup T,\ T,\ P,\ Q,\ B^u(s)).$$

Hierin ist das Axiom s der Grammatik G durch die Menge $B^u(s)$ als Axiomensystem ersetzt.

Beweis. Sei $w \in L_u$ d.h. $u \cdot w \in L$. Dann gibt es ein $f \in \mathcal{M}(S, T^*)$ mit $C(f) = u \cdot w$. Nach Lemma 4 finden wir eine Zerlegung

$$f = (1_u \times g) \circ h$$

mit $h \in \mathcal{M}(s, Z^*)$ und h u-prim. Also ist $C(h) \in u \cdot B^u(s)$, $C(g) = w$. Ist $C(h) = u \cdot v$, dann ist $v \in B^u(s)$, $D(g) = v$ und wegen $C(g) \in T^*$ also

$$g \in \mathcal{M}_G(B^u(s),\ T^*).$$

Also ist

$$w \in L(Z \cup T,\ T,\ P,\ Q,\ B^u(s)).$$

Sei nun umgekehrt $w \in L(Z \cup T,\ T,\ P,\ Q,\ B^u(s))$.

Dann gibt es eine Ableitung $f \in \mathcal{M}(s, uZ^*)$ mit $z(f) = uv$ und eine Ableitung

$$v \xrightarrow{g} w.$$

Nun ist

$$h = (1_u \times g) \circ f \in \mathcal{M}_G(s,\ T^*)$$

und also $u \cdot w \in L$, woraus $w \in L_u$ folgt.

Damit ist unser Lemma bewiesen.

Dieses Lemma rechtfertigt die Bezeichnung von $B^u(s)$ als Z-Basis von L. Wir finden als Korollar zu Lemma 2 den

Satz 1. Für $u \neq 1$ gilt $u \in L \Leftrightarrow 1 \in B^u(s)$.

Beweis. Es ist $u \in L \Leftrightarrow 1 \in L_u$. Da s in keiner Produktion rechts vorkommt folgt hieraus $1 \in B^u(s)$ oder $s \in B^u(s)$. Wegen $u \neq 1$ wäre aber die Länge $l(u \cdot s) > 1$. Da s in keiner Produktion rechts vorkommt und da $s \to us$ gelten müßte, scheidet die Alternative $s \in B^u(s)$ aus. Also gilt

$$u \in L \Rightarrow 1 \in B^u(s).$$

Der Beweis für die Inklusion in umgekehrter Richtung ist trivial.

Lemma 3. Die (t, z)-Basen sind für alle $t \in T$, $z \in Z$ reguläre Mengen.

Beweis. $\mathcal{B}^t(G, z)$ ist ein reguläres Standardereignis (regular standard event [1]) über $P \cup Q$.

Wir definieren den Homomorphismus $h: (P \cup Q)^* \to Z^*$, indem wir setzen

$$h(z_1,\ z_2 z_3) = z_3, \qquad h(z, r) = 1$$

für

$$(z_1,\ z_2 z_3) \in P \quad \text{und} \quad (z, r) \in Q.$$

Ist
$$f=(f_k\times 1_{u_k})\circ\ldots\circ(f_2\times 1_{u_2})\circ f_1,\qquad f_i\in P\quad \text{für}\quad i=1,\ldots,k-1$$
und $C(f_k)=t$, dann ist
$$h(f)=u_k\in B^t(z)\quad \text{für}\quad Q(f_1)=z.$$
Also gilt
$$B^t(z)=h(\mathscr{B}^t(G,z)).$$

Also ist $B^t(z)$ als Bild einer regulären Menge regulär. Hierzu sehe man etwa [4].

Nun liegt es nahe, das Wortproblem $u\in L(G)$ auf das Problem $1\in B^u(s)$ zurückzuführen. Die Schwierigkeit besteht darin, die endlichen Automaten, die $B^u(s)$ akzeptieren, schnell zu konstruieren. Dies gelingt, indem man einen zu $B^{ut}(z)$ gehörigen Automaten aus dem bereits für $B^u(z)$ vorhandenen Automaten gewinnt.

Ein rekursives Gleichungssystem zur Berechnung von $B^{ut}(z)$

Sei $M\subset Z^*$ und σ wie folgt definiert
$$\sigma(M)=\begin{cases}1 & \text{für}\quad 1\in M\\ \emptyset & \text{sonst.}\end{cases}$$
Mit dieser Bezeichnung gilt der

Satz 2.

1) $B^{ut}(s)=\bigcup\limits_{z\in F(B^u(s))} B^t(z)\cdot B^u_z(s).$

2) $F(B^{ut}(s))=\bigcup\limits_{z\in F(B^u(s))}[F(B^t(z))\cup\sigma(B^t(z))\cdot F(B^u_z(s))].$

Beweis. Wir zeigen zunächst 1).

Ist $f\in\mathscr{B}^{ut}(G,s)$, dann gibt es eine Zerlegung
$$f=(1_u\times g\times 1_v)\circ h$$
mit $h\in\mathscr{B}^u(s)$ und $D(g)\in Z$.

Nun ist $C(g)=t\cdot w$ mit $w\in Z^*$ und $D(g)\in F(B^u(s))$.

Also gilt $C(f)=u\cdot t\cdot w\cdot v$ mit $w\in B^t(z)$ und $v\in B^u_z(s)$.

Dann gilt
$$B^{ut}(s)\subset\bigcup_{z\in F(B^u(s))} B^t(z)\cdot B^u_z(s).$$
Sei nun umgekehrt $w\cdot v\in B^t(z)\cdot B^u_z(s)$ und $w\in B^t(z)$. Dann gibt es $f\in\mathscr{M}(s,uZ^*)$ mit $Z(f)=u\cdot z\cdot v$ und $g\in\mathscr{M}(z,tZ^*)$ mit $Z(g)=t\cdot w$. Für
$$h=(1_u\times g\times 1_v)\circ f$$
gilt dann $h\in\mathscr{M}(s,u\cdot t\cdot Z^*)$, woraus $w\cdot v\in B^{ut}(s)$ folgt. Also gilt auch
$$\bigcup_{z\in F(B^u(s))} B^t(z)\cdot B^u_z(s)\subset B^{ut}(s).$$
Aus beiden Inklusionen folgt Teil 1) des Satzes. Teil 2) folgt unmittelbar aus dem evidenten

Lemma 4. Seien $L, L' \subset Z^*$. Für $z \in Z$ gilt

1) $(L \cdot L')_z = L_z \cdot L' \cup \sigma(L) \cdot L'_z$,
$F(L \cdot L') = F(L) \cup \sigma(L) \cdot F(L')$.

Satz 1 und Satz 2 zeigen, daß wir das Wortproblem $w \in L$ auf Probleme $1 \in B_v^t(z)$ mit $t \in T$, $z \in Z$ und $v \in Z \cdot Z^*$ zurückführen können.

Aus Lemma 3 schließt man nun induktiv unter Verwendung von Satz 2 auf das

Lemma 5. $B^u(z)$ ist für $u \in T^*$ und $z \in Z$ eine reguläre Menge.

Wir konstruieren im folgenden endliche Automaten, die $B^u(z)$ akzeptieren. Bei der Konstruktion des Automaten zu $B^{ut}(z)$ verwendet man die bereits für $B^u(z)$ und $B^t(z)$ $t \in T$, $z \in Z$ gewonnenen Automaten. Für die Einfachheit der Konstruktion ist es entscheidend, daß $\{B_v^t(z) \mid v \in Z^*, z \in Z\}$ eine endliche Menge ist.

Die Bausteine des Analysealgorithmus

Sei

$$S = (Y \cup Z, Z, P, s)$$

mit $Y \cap Z = \emptyset$, $s \in Y$, $P \subset Y \times YZ \cup Y \times Y$ eine links lineare Grammatik, die keine überflüssigen Produktionen enthält, das heißt, daß jedes Zeichen $y \in Y$ durch eine Ableitung über P von s aus erreichbar ist.

Lemma 6. Sei $Q \subset Y \times \{1\}$ und $G = (Y \cup Z, Z, P, Q, s)$ eine links lineare Chomskygrammatik. Es gibt dann zu Q ein $Q' \subset Y \times Z \cup \{(s, 1)\}$, so daß G und $G' = (Y \cup Z, Z, P, Q', s)$ die gleiche Sprache definieren.

Wir beweisen dieses wohlbekannte Lemma, da wir Q' eindeutig bestimmen wollen. Zu jedem Paar von Produktionen

$$y \to y' \quad \text{und} \quad y' \to 1$$

von P nehmen wir die Produktion $y \to 1$ hinzu. Mit der so erhaltenen Grammatik verfahren wir ebenso. Dieses Verfahren setzen wir so lange fort, bis wir ein terminales Produktionssystem $\tilde{Q}$ erhalten, das sich bei diesem Verfahren nicht mehr ändert.

Nun bilden wir

$$Q' = \{(y, z) \mid \text{Es gibt } (x, 1) \in \tilde{Q} \text{ und } (y, xz) \in P\}$$

$$\cup \{(s, 1) \mid (s, 1) \in \tilde{Q}\}.$$

Man erkennt, daß P zusammen mit den terminalen Systemen $Q, \tilde{Q}$ und Q' jeweils die gleiche Sprache definiert.

Q' wird aufgrund dieser Konstruktion durch Q eindeutig bestimmt. Wir beschreiben diesen Zusammenhang durch

$$Q' = S(Q).$$

Lemma 7. Für $Q_1, Q_2 \subset Y \times \{1\}$ gilt

$$S(Q_1 \cup Q_2) = S(Q_1) \cup S(Q_2).$$

Der Beweis folgt unmittelbar aus der obigen Konstruktion. Wir bezeichnen nun die durch $G=(Y\cup Z, Z, P, Q, s)$ definierte Sprache $L(G)$ mit $L(S, Q)$. Man hat dann

Lemma 8. $L(S, Q_1\cup Q_2)=L(S, Q_1)\cup L(S, Q_2)$.

Lemma 9. Ist $Q\subset Y\times\{1\}$ und $Q'=S(Q)$, dann gilt

$$F(L(S, Q))=C(Q')-\{1\}.$$

Beweis. Da P keine überflüssigen Produktionen enthält, gilt offenbar

$$C(Q')-\{1\}\subset F(L(S, Q)).$$

Wir zeigen, daß die Inklusion auch in umgekehrter Richtung gilt. Ist $z\in F(L(S,Q))$, dann gibt es eine Ableitung

$$s \xrightarrow{f} y_1 w \xrightarrow{g_1} y z w \xrightarrow{g_2} z w.$$

Dann kommt aber nach Konstruktion die Produktion (y_1, z) in Q' vor. Also gilt auch $F(L(S, Q))\subset C(Q')-\{1\}$, was zu zeigen war.

Das folgende Lemma ist wieder unmittelbar klar.

Lemma 10. Sei $Q\subset Y\times\{1\}$.

$$1\in L(S, Q)\Leftrightarrow (s, 1)\in (S(Q).$$

Damit haben wir die Möglichkeit, das Wortproblem $1\in L(S, Q)$ durch die Berechnung von $S(Q)$ zu lösen. Da uns $S(Q)$ auch $F(L(S, Q))$ mitliefert, kann man hoffen, unser Problem $1\in B^u(s)$ mit diesem Algorithmus angehen zu können. Dies wird weiter begründet durch das unten stehende Lemma 11.

Zunächst wollen wir aus Grammatiken $G_0, \ldots, G_n$ eine neue Grammatik G zusammensetzen. Hierzu nehmen wir an, daß das Terminalalphabet in allen Grammatiken Z ist, und daß die Alphabete Y_i paarweise fremd sind. Es sei $A=\{s_1, \ldots, s_n\}$, worin s_i das Axiom von G_i ist und $\eta: Z\to A$ eine Abbildung. Weiter sei $Q_i\subset Y_i\times\{1\}$ für $i=1, \ldots, n$ und $Q_0\subset Y_0\times Z\cup\{(s, 1)\}$.

Lemma 11. Es gelten die eben vereinbarten Bezeichnungen und es sei $G=(Y\cup Z, Z, P, Q, s_0)$ mit

$$Y=\bigcup_{i=0}^{n} Y_i, \qquad P=\bigcup_{i=0}^{n} P_i\cup\{(y, \eta(z))\mid (y, z)\in S(Q_0)\}$$

$$Q=\bigcup_{i=1}^{n} Q_i.$$

Hieraus folgt

$$L(G)=\bigcup_{\eta(z)=s_i} L(S_i, Q_i)\cdot L(S_0, Q_0).$$

Beweis. 1) Wir beweisen, daß die linke Seite in der rechten Seite enthalten ist. Sei also $w\in L(G)$, dann gibt es eine Ableitung

$$s\xrightarrow{f} w.$$

Da $Q \cap Q_0 = \emptyset$, stammt die in f enthaltene terminale Produktion aus Q_1 oder Q_2 oder ... oder Q_n. O.B.d.A. können wir annehmen, daß diese Produktion aus Q_1 ist. Dann läßt sich f aber wie folgt zerlegen.

$$s \xrightarrow{f_1} y \cdot u \xrightarrow{f_2} \eta(z) \cdot u \xrightarrow{f_3} w,$$

worin $f_2 = (y, \eta(z)) \times 1_u$ ist. Dann gilt $u \in L_z(S_0, Q_0)$ und $v \in L(S_1, Q_1)$, wo $w = v \cdot u$ ist. Hieraus folgt, daß $L(G)$ in der rechten Seite der obigen Gleichung enthalten ist.

2) Die rechte Seite ist in $L(G)$ enthalten. Sei also $w \in Z^*$ in der rechten Seite enthalten. Dann gibt es eine Zerlegung $w = uv$, so daß $u \in L(S_i, Q_i)$ für geeignetes i und $zv \in L(S_0, Q_0)$ für geeignetes $z \in Z$. Dann gibt es eine Ableitung f in G_0 und g in G_i mit

$$s \xrightarrow{f} y \cdot v \xrightarrow{(y,z)} z \cdot v, \qquad \eta(z) \xrightarrow{g} u$$

und also

$$s \xrightarrow{h} u \cdot v$$

für $h = (g \times 1_v) \circ ((y \to \eta(z)) \times 1_v) \circ f$. h ist aber eine Ableitung in G, womit auch diese Richtung des Satzes bewiesen ist.

Lemma 12. Die Bezeichnungen und Voraussetzungen sind die gleichen wie in Lemma 6. Es gilt dann

$$S(Q) = \bigcup_{i=1}^{n} S_i(Q_i) \cup S_0(Q')$$

mit

$$Q' = \{(y, 1) \mid (y, s_i) \in P \text{ und } (s_i, 1) \in S_i(Q_i)\}.$$

Beweis. Wenden wir den ersten Teil unseres Algorithmus zur Berechnung von $S(Q)$ an, dann wird in diesem Teil Q' erzeugt. Der zweite Teil dieses Algorithmus wirkt nur auf Produktionen der Form $y \to yz$, d.h. nur die „Komponenten" G_i von G. Hieraus folgt die Behauptung des Satzes.

Eine Plausibilitätsbetrachtung

Sei nun K irgendein Komplexitätsmaß, das die Berechnung von $S(Q)$ durch unseren Algorithmus mißt. Es gilt dann

$$K(S(Q)) = K(S(Q_0)) + \sum_{i=1}^{n} K(S_i(Q_i)) + K(Q'),$$

worin $K(Q')$ die Komplexität der Berechnung von Q' ist. $K(Q')$ hängt offensichtlich von G_0 ab.

Betrachten wir die uns interessierenden Sprachen $B^u(s)$, dann erhalten wir mittels der rekursiven Darstellung von $B^{t \cdot u}(s)$ für die Komplexität $K(1 \in B^{tu}(s))$ des Wortproblems $1 \in B^{tu}(s)$, wenn K_0 eine Schranke für die Komplexität von $S(Q)$ bezüglich $B^t(z)$, $t \in T$, $z \in Z$ darstellt. Wir setzen $c = K_0 \operatorname{card} Z$ und haben für $l(ut) = N$

$$K(1 \in B^{tu}(s)) \leqq c \cdot \sum_{i=1}^{N} i + \sum_{i=1}^{N} K(Q_i'),$$

das heißt

$$K(1 \in B^{tu}(s)) \leqq c \cdot \binom{N+1}{2} + \sum_{i=1}^{N} \sum_{j=1}^{i} K(Q_j')$$

Wir können also erwarten, daß $K(1 \in B^u)$ höchstens quadratisch mit $1(n)$ wächst, wenn $K(Q_i')$ beschränkt ist. Wächst $K(Q_j')$ jedoch linear, dann können wir für $K(1 \in B^u(s))$ nur eine kubische Schranke entwickeln. Q_i' wird durch eine Abbildung $\eta_i: Y_i \to Z$ beschrieben. Y_i wächst aber mit u linear. Deshalb scheint auf den ersten Blick das bis hierhin skizzierte Verfahren eine kubische Komplexität zu besitzen.

Die detaillierte Ausführung dieser Skizze erfolgt im restlichen Teil der Arbeit.

Zunächst definieren wir einen endlichen Automaten, der im wesentlichen $S(Q)$ für die Sprachen $B^t(z)$ berechnet. Aus diesem Automaten setzen wir ein Automatenarray zusammen, das $S(Q)$ für $B^u(s)$ berechnet. Schließlich simulieren wir das Array auf einer RAM.

Wir betrachten nun spezielle Systeme S, die wir aus unserer eingangs gegebenen Grammatik

$$G = (Z \cup T, T, P, Q, s)$$

ableiten. Hierzu verdoppeln wir das Alphabet Z. Das zweite Exemplar wird mit Y bezeichnet. Wir setzen voraus $Y \cap Z = \emptyset$. Zwischen Y und Z zeichnen wir eine Bijektion aus, die wir durch $z = z(y)$ bzw. $y = y(z)$ ausdrücken.

Nun definieren wir

$$P_Y = \{ (y(z), y(z_1) z_2) \mid (z, z_1, z_2) \in P \},$$

$$Q_t = \{ (y(z), 1) \mid (z, t) \in Q \} \quad \text{für} \quad t \in T.$$

Nun bilden wir

$$S_{t,z} = (Y \cup Z, Z, P_Y, y(z))$$

und hieraus durch Weglassen der überflüssigen Produktionen und Alphabetelemente

$$S^{t,z} = (Y^{t,z} \cup Z^{t,z}, Z^{t,z}, P_Y^{t,z}, y(z)).$$

Setzt man

$$Q^{t,z} = \{(y, 1) \mid (z(y), t) \in Q, \ y \in Y^{t,z}\},$$

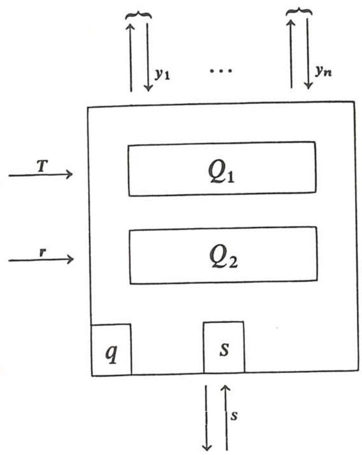
dann hat man

$$L(S^{t,z}, Q^{t,z}) = B^t(z).$$

Zu diesen Systemen $S^{t,z}$ konstruieren wir nun den endlichen Automaten, der $S^{t,z}(Q^{t,z})$ berechnet.

Definition des Automaten $\mathscr{A}(G, t, z)$

Zur Vereinfachung unserer Sprechweise unterscheiden wir bei unserem Automaten zwischen dem *unteren Eingang*, den *oberen Eingängen* und den *seitlichen Eingängen*. Zu jedem oberen und dem unteren Eingang gehört genau ein *oberer* bzw. *unterer Ausgang*. Wir haben zwei seitliche Eingänge, den T-Eingang und den r-Eingang. Der Signalvorrat des T-Eingangs besteht gerade aus den Elementen der Menge T. Alle übrigen Eingänge und alle Ausgänge sind binär. Der *Zustand* des Automaten ist gegeben durch den Inhalt zweier Register Q_1 und Q_2 und eines binären Speicherlementes s. Der r-Eingang dient der Umspeicherung von Q_2 nach Q_1 und dem Löschen von Q_2. In Q_1 und Q_2 werden terminale Produktionssysteme Q abgespeichert, in s die eventuell vorhandene Produktion $s \to 1$.



q sorgt dafür, daß obere Ausgangssignale nur einen Takt nach der Anweisung $r := 1$ erscheinen können.

Es sei $\mathscr{A}(G, t, z)$ ein endlicher Automat mit dem *Eingabealphabet* $\mathscr{X}$, dem *Ausgabealphabet* $\mathscr{Y}$, der *Zustandsmenge* $\mathscr{R}$, dem Anfangszustand a und der *Überführungsfunktion* δ und der Ausgabefunktion λ.

$$\delta : \mathscr{X} \times \mathscr{R} \to \mathscr{R}, \qquad \lambda : \mathscr{R} \to \mathscr{Y}$$

Die Mengen $\mathscr{X}, \mathscr{Y}, \mathscr{R}$ sind wie folgt definiert:

$$\mathscr{X} = T \times \mathscr{X}_1, \qquad \mathscr{X}_1 = \{\xi : (Y \cup \{r, s\}) \to \{0, 1\}\},$$

$$\mathscr{Y} = \{\eta : (Y \cup \{s\}) \to \{0, 1\}\},$$

$$\mathscr{R}' = \wp(Y \times Z) \times \wp(Y \times Z) \times \{\zeta : \{s, q\} \to \{0, 1\}\}$$

und

$$\mathscr{R} = \mathscr{R}' \cup \{a\} \quad \text{mit} \quad a \notin \mathscr{R}'.$$

Hierin bezeichnet $\wp M$ die Potenzmenge von M und $s = s(z)$ ein Symbol das wir später auch mit $y'(z)$ oder $x'(s)$ bezeichnen werden, um eine Korrespondenz mit dem Axiom von $S^{t,z}$ zu betonen.
$\xi(y_1)$ ist also das Eingangssignal, das dem Eingang y_1, $\xi(s)$ das Signal, das dem Eingang s zugeordnet ist. Entsprechend ist $\lambda(y_5)$ das am Ausgang y_5 vorhandene Ausgangssignal, und $\zeta(s)$ der Inhalt der Speicherzelle s.

Wir definieren zunächst δ. Dies geschieht zuerst für den Anfangszustand a.

$$\delta(t', \xi, a) = \begin{cases} a & \text{für} \quad t' \neq t \quad \text{oder} \quad \xi(s) = 0 \\ (\phi, Q^{t,z}, \zeta) & \text{sonst} \end{cases}$$

mit

$$\zeta(s) = 1 \Leftrightarrow (y(z), 1) \in Q^{t,z}$$

und

$$\zeta(q) = 1.$$

Sei nun $(Q_1, Q_2, \zeta) \in R$ nicht der Anfangszustand.

Wir setzen

$$(Q_1', Q_2', \zeta') = \delta(t', \xi, Q_1, Q_2, \zeta)$$

mit

$$Q_1' = \begin{cases} Q_1 & \text{für} \quad \xi(r) = 0 \\ Q_2 & \text{für} \quad \xi(r) = 1 \end{cases}$$

und

$$Q_2 = \begin{cases} \emptyset & \text{für} \quad \xi(r) = 1 \\ Q_2 \cup \left(S(Q) - \{(y(z), 1)\}\right) & \text{für} \quad \xi(r) = 0. \end{cases}$$

Hierin ist

$$Q = \{(y, 1) \mid \xi(y) = 1, y \in Y\}.$$

Wir setzen weiter unter Verwendung dieser Definition

$$\zeta'(s) = \begin{cases} 1 & \text{für} \quad (y(z), 1) \in S(Q) \\ 0 & \text{sonst} \end{cases}$$

und

$$\zeta'(q) = 1 \Leftrightarrow \xi(r) = 1.$$

q wird also durch $\xi(r) := 1$ auf 1 gesetzt und durch $\xi(r) = 0$ auf 0, falls $\mathscr{A}(G, t, z)$ nicht im Anfangszustand ist.

Es bleibt noch λ zu definieren.

Wir setzen

$$\eta(y) = 1 \Leftrightarrow (y, z) \in Q_1 \quad \text{und} \quad \zeta(q) = 1 \quad \text{für} \quad y \in Y$$

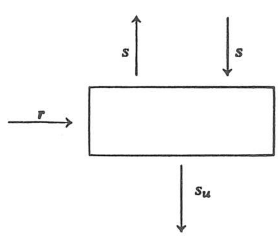
und

$$\eta(s) = 1 \Leftrightarrow \zeta(s) = 1.$$

$\mathscr{A}(G, t, z)$ gibt also nur für $\zeta(q) = 1$ Signale nach oben ab.

Definition des Automaten $a(G)$

Zum Start des später zu definierenden Arrays verwenden wir den Automaten $a(s)$.



$a(G)$ besitzt die drei Zustände $\sigma_0, \sigma_1, \sigma_2$.

$a(G)$ besitzt einen oberen und einen unteren Ausgang.

$a(G)$ besitzt einen seitlichen und einen oberen Eingang.

Die Bezeichnung entnehme man der Figur. σ_0 ist der Anfangszustand von $a(G)$. Die Funktionen δ und λ von $a(G)$ sind wie folgt definiert:

$$\begin{aligned} &\delta(r, s, \sigma_0) = \sigma_1, \\ &\left.\begin{aligned} \delta(0, 1, \sigma) &= \sigma_2 \\ \delta(1, s, \sigma) &= \sigma_1 \end{aligned}\right\} \quad \text{für } \sigma \in \{\sigma_1, \sigma_2\}, \\ &\delta(r, 0, \sigma_1) = \sigma_2. \end{aligned}$$

Der Automat geht also aus seinem Anfangszustand σ_0 in jedem Fall nach σ_1 über. Ist $\xi(r) = 0$, dann führt $\xi(s) = 1$ den Automaten nach σ_2 über. In diesem Zustand merkt sich der Automat, daß von oben eine 1 eingetroffen ist. Durch $\xi(r) = 1$ wird er in seinen „Grundzustand" σ_1 gesetzt, indem er auf Signale von oben wartet. Wir definieren λ:

$$\lambda(\sigma) = (\eta(s), \eta(s_n))$$

mit

$$\begin{aligned} \lambda(\sigma_0) &= (1, \varepsilon), \qquad \varepsilon = 1 \Leftrightarrow 1 \in L(G). \\ \lambda(\sigma_1) &= (0, 0) \\ \lambda(\sigma_2) &= (0, 1). \end{aligned}$$

Der Automat $a(G)$ zeigt also im Anfangszustand σ_0 an, ob das leere Wort in $L(G)$ liegt. Sonst zeigt er an, ob seit dem letzten Signal $\zeta(r) = 1$ von oben eine 1 eingetroffen ist.

Das Automatenarray $\mathscr{A}(G)$

Wir ordnen G eine Matrix $\mathscr{A}(G)$ mit unendlich vielen Zeilen und endlich vielen Spalten zu. Die Zeilen werden mit $0, 1, 2, \ldots$ durchnumeriert, die Spalten mit $(t, z) \in T \times Z$. Die unterste Zeile trage den Index 0, d.h., daß wir uns die Matrix als noch oben unendlich vorstellen. Die Gitterpunkte (i, t, z) der Matrix sind für $i = 1, 2, \ldots$ mit unseren Automaten $\mathscr{A}(G, t, z)$ besetzt, und zwar sitzt auf dem Punkt (i, t, z) ein Exemplar des Automaten $\mathscr{A}(G, t, z)$.

An den Stellen $(0, t, z)$ sitzt der Automat $a(s)$; s ist das Axiom von G.

Aus Gründen der Übersichtlichkeit drücken wir die Abhängigkeit von G nicht mehr aus. Der an der Stelle (i, t, z) sitzende Automat $\mathscr{A}(G, t, z)$ bzw. $a(s)$ wird also mit $\mathscr{A}(i, t, z)$ bezeichnet. $\mathscr{A} = A(G)$ heißt die zu G gehörige *Automatenmatrix*. Die Zustände, Ein- und Ausgänge der Automaten $\mathscr{A}(i, t, z)$ werden mit (i, t, z) indiziert. Die Menge der *Eingänge* (*Ausgänge*) von $\mathscr{A}$ besteht in der Vereinigung der Menge der *Eingänge* (*Ausgänge*) der Automaten $A(i, t, z)$. Entsprechend ist die Menge der *oberen* (*unteren, linken*) Eingänge (Ausgänge) die Vereinigung der entsprechenden Mengen der Automaten $A(i, t, z)$. $I(\mathscr{A})$ bezeichne die Menge der Eingänge von $\mathscr{A}$ und $O(\mathscr{A})$ die Menge der Ausgänge von $\mathscr{A}$.

Ein Eingangssignal für $\mathscr{A}$ ist eine Abbildung der Menge der Eingänge in $\{0, 1\} \cup T$, die der Nebenbedingung genügt, daß das Bild der t-Eingänge in T liegt und das der anderen Eingänge in $\{0, 1\}$. Entsprechend verstehen wir das *Ausgangssignal* und den *Zustand* von $\mathscr{A}(G)$.

Ein Ausgang, der $y \in Y$ zugeordnet ist wird y-Ausgang genannt und als Ausgang von $\mathscr{A}(i, t, z)$ mit $y(i, t, z)$ bezeichnet. Handelt es sich um einen Eingang, dann schreiben wir y' anstelle von y.

Die Unterscheidung „oberer" und „unterer" Ein- oder Ausgang bringen wir in der Notation nicht zum Ausdruck.

Wir ergänzen nun die Matrix $\mathscr{A}$ durch eine Verdrahtung $\mathscr{E}$. $\mathscr{E}$ enthält also „Verbindungsdrähte" zwischen Ein- und Ausgängen von $\mathscr{A}$.

Formal mathematisch ist

$$\mathscr{E} \subset O(\mathscr{A}) \times I(\mathscr{A}).$$

$\mathscr{E}$ heißt eine zulässige Verdrahtung von $\mathscr{A}$, wenn (E1), ..., (E6) gilt.

(E1) $\mathscr{E}$ ist endlich.

(E2) Ist $(a, b) \in \mathscr{E}$, dann ist entweder a ein oberer Ausgang und b ein unterer Eingang, oder a ein unterer Ausgang und b ein oberer Eingang.

(E3) Ist $(a, b) \in \mathscr{E}$ und ist $a = x(i, t, z)$ ein unterer Ausgang und $b = y'(j, r, z')$ ein oberer Eingang, dann ist $i > j$, $x = y$ und $x = z$. (Letzteres folgt an sich schon aus der Definition von $\mathscr{A}(G, t, z)$).

(E4) Zu jedem $(a, b) \in \mathscr{E}$, $a = x(i, t, z)$, $b = y'(j, r, z')$ gibt es genau ein $(b', a') \in \mathscr{E}$ mit $a' = x'(i, t, z)$ und $b' = y(j, r, z')$. Hierin ist a' ein oberer (unterer) Eingang, wenn a ein oberer (unterer) Ausgang ist und b' ein unterer (oberer) Ausgang, wenn b ein unterer (oberer) Eingang ist. Bezeichnung: $(b', a') = (a, b)^{-1}$.

(E5) Ist $(a, b) \in \mathscr{E}$, $a = x(i, t, z)$, dann gibt es eine Folge $(a_j, b_j) \in \mathscr{E}$, $j = 1, \ldots, m$ mit

$$a_j = x_j(i_{j-1}, t_j, x_{j-1}), \qquad b_j = x_j'(i_j, t_{j+1}, x_j)$$

und es ist

$$x_0 = s, \qquad i_0 = 0, \qquad i_{j-1} < i_j, \qquad b = x_m'(i, t, z).$$

Hierin ist s das Axiom von G. (Wir bezeichnen (E5) als Zusammenhangsaxiom.)

(E6) Es gibt ein ausgezeichnetes Element $t_0 \in T$ mit folgender Eigenschaft: Für jedes $(a, b) \in \mathscr{E}$ mit $a = x(0, t, z)$ und $b = x'(i, r, z')$ gilt $t = t_0$, $z = s$ und $i = 1$. (Wir bezeichnen (E6) als Verankerungsaxiom.)

Das Axiom E6 besagt, daß wir von den Automaten der 0-ten Zeile genau einen verwenden und daß die anderen an sich überflüssig sind. Die von a ausgehenden Drähte gehen auch nur bis zur Zeile 1.

Definition. Ein Eingang a von $\mathscr{A}$ heißt *freier Eingang* von $(\mathscr{A}, \mathscr{E})$, wenn er kein T- oder r-Eingang ist, und wenn es keinen Ausgang b von $\mathscr{A}$ gibt mit $(b, a) \in \mathscr{E}$.

Wir legen nun die Arbeitsweise von $(\mathscr{A}, \mathscr{E})$ fest. Hierzu definieren wir zunächst mit $B = \{0, 1\}$:

(A1) $T \times B$ ist die Menge der Eingangssignale für $(\mathscr{A}, \mathscr{E})$. Durch (A2) bis (A4) werden aus dem Eingangssignal (t, r) für $(\mathscr{A}, \mathscr{E})$ und dem momentanen Zustand ζ von $(\mathscr{A}, \mathscr{E})$ die Eingangssignale für die Automaten $\mathscr{A}(i, t, z)$ abgeleitet.

(A2) Ist (t, r) das Eingangssignal für $(\mathscr{A}, \mathscr{E})$ im Zustand ζ, dann ist (t, r) auch seitliches Eingangssignal für $\mathscr{A}(i, t, z)$ für $i > 0$ und r für $A(0, t, z)$ im Zustand $\zeta(i, t, z)$.

(A3) Ist a freier Eingang von $(\mathscr{A}, \mathscr{E})$ dann liegt an a stets das Signal 0 an.

(A4) Ist b Eingang von $\mathscr{A}$ und gibt es ein a mit $(a, b) \in \mathscr{E}$, dann gilt für das Eingangssignal $\xi(b)$

$$\xi(b) = \bigcup_{(a,b) \in E} \eta(a),$$

worin $\eta(a)$ das am Ausgang a anstehende Ausgangssignal ist und $\bigcup$ die logische Summe ist. Das heißt, daß wir im Falle, daß mehrere Ausgänge $a_1, \ldots, a_k$ auf den Eingang b geschaltet sind, die Signale $\eta(a_1), \ldots, \eta(a_k)$ nach Rechenregel

$$0 \cup 0 = 0, \qquad 0 \cup 1 = 1 \cup 0 = 1 \cup 1 = 1$$

überlagern.

Damit sind für jedes Eingangssignal (t, r) für $(\mathscr{A}, \mathscr{E})$ und jeden Zustand ζ von $(\mathscr{A}, \mathscr{E})$ die momentanen Eingangssignale und Zustände für jeden Automaten $\mathscr{A}(i, t, z)$ definiert.

(A 5) Es gilt

$$\zeta' = \delta(t, r, \zeta) \quad \text{für} \quad (t, r) \in T \times B$$

und ζ, ζ' Zustände von $(\mathscr{A}, \mathscr{E})$, genau dann, wenn $\zeta'(i, t, z)$ Folgezustand von $\xi(i, t, z)$ des Automaten $\mathscr{A}(i, t, z)$ bei den durch (A 2) bis (A 4) festgelegten Eingangssignalen ist.

(A 6) Das Ausgangssignal des ausgezeichneten Automaten $\mathscr{A}(0, t_0, s)$ ist das Ausgangssignal von $(\mathscr{A}, \mathscr{E})$.

(A 7) ζ ist Anfangszustand von $(\mathscr{A}, \mathscr{E})$, wenn $\zeta(i, t, z)$ Anfangszustand von $\mathscr{A}(i, t, z)$ für alle (i, t, z) ist.

Nun klären wir durch einige Hilfssätze die Eigenschaften des durch (A 1), …, (A 7) definierten Automaten $(\mathscr{A}, \mathscr{E})$. Wir setzen im folgenden stets voraus, daß $\mathscr{E}$ für $\mathscr{A}$ zulässig ist, ohne dies jedesmal festzustellen.

Lemma 13. Ist der untere Eingang $x'(i, t, x)$ von $\mathscr{A}(i, t, x)$ freier Eingang von $(\mathscr{A}, \mathscr{E})$, dann sind auch die oberen Eingänge von $\mathscr{A}(i, t, x)$ frei.

Beweis. Folgt unmittelbar aus dem Zusammenhangsaxiom.

Lemma 14. Ist der untere Eingang von $\mathscr{A}(i, t, z)$ freier Eingang von $(\mathscr{A}, \mathscr{E})$ und ist $\mathscr{A}(i, t, z)$ in seinem Anfangszustand, dann bleibt $\mathscr{A}(i, t, z)$ stets in seinem Anfangszustand.

Beweis. Folgt unmittelbar aus (A 3).

Korollar zu Lemma 14. Ist ζ_0 der Anfangszustand von $(\mathscr{A}, \mathscr{E})$, dann ist $(\mathscr{A}, \mathscr{E}, \zeta_0)$ als initialer Automat bis auf Isomorphie ein endlicher Automat.

Definition. Die *Höhe* $H(\mathscr{A}, \mathscr{E})$ ist definiert durch

$$H(\mathscr{A}, \mathscr{E}) = \max\{i \mid x'(i, t, z) \text{ ist nicht freier Eingang von } (\mathscr{A}, \mathscr{E})\}.$$

Lemma 15. Ist $H(\mathscr{A}, \mathscr{E}) = N$, dann gilt für alle $k > N$, $(t, r) \in T \times B$ und $l(w) = k$.

$$\delta(w, r^k, \zeta) = \text{constant}.$$

Beweis. Ist $r = 1$, dann gilt die Behauptung schon für $k > 2$, da das erste Signal $r = 1$ bewirkt, daß der Inhalt der Q_2-Register in die zugehörigen Q_1-Register gebracht wird und Q_2 anschließend auf $\emptyset$ gesetzt wird. Das zweite $r = 1$-Signal wiederholt diese Operation, so daß nun für alle Register gilt $Q_1 = Q_2 = \emptyset$. Hieraus ergibt sich daß alle unteren und oberen Ausgänge eine 0 anzeigen. Weitere Signale können diesen Zustand nicht mehr verändern.

Wir betrachten den Fall $r = 0$.

Nach Definition von $\mathscr{A}(G, t, z)$ tritt nur dann ein oberes Ausgangssignal $\neq 0$ auf, wenn unmittelbar vorher das Signal $r = 1$ angelegen hatte. Das heißt, daß im Zustand $\zeta_1 = \delta(w, 0^k, \zeta)$ an sämtlichen oberen Ausgängen von $\mathscr{A}$ eine 0 ansteht für $k > 1$ und $w \in T^*$, $1(w) = k$.

Der Zustand eines Automaten $\mathscr{A}(i, t, z)$ von $(\mathscr{A}, \mathscr{E})$ kann also für $k > 1$ nur durch obere Eingangssignale geändert werden. Ein Automat gibt aber höchstens dann ein Eingangssignal nach unten ab, wenn er ein Eingangssignal $\neq 0$ erhält. Da für $k > 1$ keine unteren Eingangssignale $\neq 0$ mehr auftreten und $H(\mathscr{A}, \mathscr{E}) = N < \infty$ ist, wird in jedem Takt der maximale Index i der Automaten $\mathscr{A}(i, t, z)$, die ein Signal nach unten abgeben um 1 erniedrigt. Also nach spätestens $N + 1$ Schritten gibt kein Automat $\mathscr{A}(i, t, z)$ ein Signal $\neq 0$ ab. Das heißt aber, daß für $k > N$ der Zustand

$$\zeta_k = \delta(w, 0^k, \zeta) = \zeta_{N+1}$$

ist, was zu beweisen war.

Definition. ζ heißt *stabiler* Zustand von $(\mathscr{A}, \mathscr{E})$, wenn $\zeta = \delta(t, 0, \zeta)$ für $t \in T$.

Definition. Ist ζ ein Zustand von $(\mathscr{A}, \mathscr{E})$ und ist $\zeta' = \delta(tw, 0^{N+1}, \zeta)$ mit $l(w) = N$, dann nennen wir ζ' den zu (t, ζ) gehörigen *stabilen Zustand* und schreiben

$$\zeta' = \varrho(t, \zeta).$$

Lemma 16. ϱ ist wohldefiniert.

Beweis. Es bleibt aufgrund von Lemma 15 nur zu zeigen, daß $\delta(t_1 t_2, 0^2, \zeta)$ von t_2 nicht abhängt. Dies folgt aber daraus, daß der Folgezustand von $\mathscr{A}(i, t, z)$ nur dann von t abhängt, wenn der untere Eingang von $\mathscr{A}(i, t, z)$ eine 1 liefert. Dies ist aber höchstens nach dem Signal $r = 1$ der Fall, woraus die Behauptung folgt.

Lemma 17. $(\mathscr{A}, \mathscr{E})$ ist genau dann in einem stabilen Zustand ζ, wenn alle Ausgänge von $\mathscr{A}$ außer etwa der von $\mathscr{A}(0, t_0, s)$ das Signal 0 liefern.

Beweis. 1. Alle Ausgänge seien 0. Ein Automat $\mathscr{A}(i, t, z)$ erhält in diesem Fall als Eingangssignale nur Signale 0, außer über den T-Eingang. Dann ändert sich der Zustand von $\mathscr{A}(i, t, z)$ und damit der von $(\mathscr{A}, \mathscr{E})$ nicht.

2. Sei $\zeta' = \delta(t, 0, \zeta)$ und $\zeta' = \zeta$. Wir zeigen, daß dann im Zustand ζ alle Ausgangssignale der Automaten $\mathscr{A}(i, t, z)$ gleich 0 waren.

Dies folgt unmittelbar aus der Definition von $\mathscr{A}(i, t, z)$, denn nachdem $\mathscr{A}(i, t, z)$ eine 1 nach oben abgegeben, hat wird $q = 0$ gesetzt. Hat $\mathscr{A}(i, t, z)$ eine 1 nach unten abgegeben, wird $s(i, t, z) = 0$ gesetzt, es sei denn, daß durch ein oberes Eingangssignal $s(i, t, z)$ auf 1 gehalten wird. Da $\mathscr{E}$ endlich ist, folgt aus der Betrachtung eines obersten $\mathscr{A}(i, t, z)$, das ein Signal nach unten abgegeben hat, die Behauptung.

Die zu $(\mathscr{A}(G), \mathscr{E})$ gehörigen Grammatiken

Wir ordnen nun jeden Automatenarray $(\mathscr{A}, \mathscr{E})$ in Verbindung mit jedem seiner Zustände ζ zwei Grammatiken $Gr^1(\mathscr{A}, \mathscr{E}, \zeta)$ und $Gr^2(\mathscr{A}, \mathscr{E}, \zeta)$ zu. Dies geschieht, indem wir zunächst jeden Automaten $\mathscr{A}(i, t, z)$ in Verbindung mit jedem seiner Zustände $\zeta(i, t, z)$ eine Grammatik zuordnen. Danach setzen wir diese Grammatiken durch die Konstruktion eines trivialen Produktionssystems $\mathcal{S}(\mathcal{E})$ zusammen.

Sei also

$$\zeta(i, t, z) = \left(Q_{1,i}^{t,z}(\zeta), Q_{2,i}^{t,z}(\zeta), s_i^{t,z}(\zeta), q_i^{t,z}(\zeta)\right)$$

der Zustand von $\mathcal{A}(i, t, z)$. Wir setzen für $i > 0$

$$Gr^1(i, t, z, \zeta) = \left(Y_i^{t,z} \cup Z, Z, P_i^{t,z}, Q_{1,i}^{t,z}(\zeta), y_i^{t,z}(z)\right)$$

und

$$Gr^2(i, t, z, \zeta) = \left(Y_i^{t,z} \cup Z, Z, P_i^{t,z}, \tilde{Q}_{2,i}^{t,z}, y_i^{t,z}(z)\right)$$

mit

$$\tilde{Q}_{2,i}^{t,z}(\zeta) = \begin{cases} Q_{2,i}^{t,z}(\zeta) & \text{falls} \quad s_i^{t,z}(\zeta) = 0 \\ Q_{2,i}^{t,z}(\zeta) \cup \left\{\left(y_i^{t,z}(z), 1\right)\right\} & \text{falls} \quad s_i^{t,z}(\zeta) = 1. \end{cases}$$

Hierin ist $P_i^{t,z}$ bis auf die Indizierung nach i das nichtterminale Produktionssystem $P^{t,z}$ von $G^{t,z}$.

Aufgrund der Indizierung ist

$$Y_i^{t,z} \cap Y_j^{r,x} = \emptyset \quad \text{für} \quad (i, t, z) \neq (j, r, x).$$

Wir definieren unter Verwendung der Bezeichnung $Y_0^{t,s} = \{s\}$ das angekündigte Produktionssystem $\mathcal{S}(\mathcal{E})$ mit dem wir die Grammatiken $Gr^j(i, t, z, \zeta)$ zu *einer* Grammatik verbinden.

$$\mathcal{S}(\mathcal{E}) = \{(y_1, y_2) \mid y_1 \in Y_i^{t,z}, y_2 \in Y_j^{r,z}, i < j, (y_1, y_2') \in \mathcal{E}\}.$$

Das Produktionssystem enthält also die Produktion

$$y_1 \to y_2$$

genau dann, wenn der obere Ausgang y_1 mit dem unteren Eingang y_2' verbunden ist.

Wir bilden nun die zusammengesetzten Grammatiken.

Sei

$$Y(\mathcal{E}) = \bigcup_{\substack{\exists \\ (a,b) \in \mathcal{E}}} \bigcup_{(b' \in Y_j^{t,z})} Y_i^{t,z},$$

$$P(\mathcal{E}) = \bigcup_{Y_j^{t,z} \subset Y(\mathcal{E})} P_i^{t,z} \cup \mathcal{S}(\mathcal{E}),$$

$$Q_2(\zeta) = \bigcup_{\substack{i,t,z \\ i>0}} \tilde{Q}_{2,i}^{t,z}(\zeta) \cup \begin{cases} \emptyset & \text{für} \quad \zeta(0, t_0, s) \neq \sigma_3 \\ \{(s, 1)\} & \text{für} \quad \zeta(0, t_0, s) = \sigma_3. \end{cases}$$

Nun setzen wir

$$Gr^{(j)}(\mathcal{A}, \mathcal{E}, \zeta) = \left(Y(\mathcal{E}) \cup Z, Z, P(\mathcal{E}), Q_j(\zeta), s\right).$$

Die folgenden Lemmata klären den Zusammenhang zwischen den zu $Gr^{(j)}(\mathcal{A}, \mathcal{E}, \zeta)$ gehörigen Sprachen in Abhängigkeit von j und ζ.

Lemma 18. Ist $\zeta' = \varrho(r, \zeta)$, dann ist

$$Gr^1(\mathcal{A}, \mathcal{E}, \zeta) = Gr^1(\mathcal{A}, \mathcal{E}, \zeta').$$

Beweis. Der Inhalt der Register $Q_1(i, t, z)$ mit $i > 0$ ändert sich höchstens durch die Eingabe eines Signals (t, r) mit $r = 1$. Gr^1 hängt aber nur von $Q_1(i, t, z)$ mit $i > 0$ ab, woraus die Behauptung folgt.

Lemma 19. Ist $\zeta' = \delta(1, t, \zeta)$ und ist ζ stabil, dann gilt

$$Gr^1(\mathscr{A}, \mathscr{E}, \zeta') = Gr^2(\mathscr{A}, \mathscr{E}, \zeta) - \{(s, 1)\}.$$

Beweis. Nach Lemma 17 sind alle Ausgänge von $\mathscr{A}(i, t, z)$ für $i > 0$ gleich 0. Also ist für alle (i, t, z) mit $i > 0$

$$\tilde{Q}^{t,z}_{2,i}(\zeta) = Q^{t,z}_{2,i}(\zeta).$$

Da das Signal $r = 1$ den Inhalt der Q_2-Register nach Q_1 bringt, gilt also die Behauptung des Satzes.

Der folgende Hilfssatz verallgemeinert Lemma 12

Lemma 20. Ist $\zeta' = \delta(t, 0, \zeta)$ und $\zeta = \delta(r, 0, \zeta_1)$, dann gilt

$$L\left(Gr^2(\mathscr{A}, \mathscr{E}, \zeta)\right) = L\left(Gr^2(\mathscr{A}, \mathscr{E}, \zeta')\right).$$

Beweis. Wegen $\zeta = \delta(r, 0, \zeta_1)$ sind im Zustand ζ alle oberen Ausgangssignale gleich 0. Also kann sich der Zustand von $\mathscr{A}$ nur durch von oben kommende Eingangssignale ändern.

Gibt $\mathscr{A}(i, t, z)$ nach unten an die Automaten $\mathscr{A}(i_1, t_1, z_1), \ldots, \mathscr{A}(i_k, t_k, z_k)$ Signale 1 ab, dann ändern sich die Grammatiken Gr^2 wie folgt:

Aus $Gr^2(i, t, z, \zeta)$ wird die Produktion

$$y^{t,z}_i(z) \to 1$$

herausgenommen und an die Grammatiken

$$Gr^2(i_1, z_1, z_1, \zeta), \ldots, Gr^2(i_k, t_k, z_k, \zeta)$$

weitergegeben.

Die zugehörigen Automaten rechnen in einem Schritt die zu diesen Grammatiken gehörigen neuen terminalen Produktionssysteme aus. Man erhält die neuen Grammatiken

$$Gr^2(i_1, t_1, z, \zeta'), \ldots, Gr^2(i_k, t_k, z_k, \zeta').$$

Da $Gr(A, \mathscr{E}, \zeta)$ die Produktionen

$$y(i_j, t_j, z_1) \to y'(i, t, z) \quad \text{für} \quad j = 1, \ldots, k$$

enthält, ändert sich die durch $Gr^2(\mathscr{A}, \mathscr{E}, \zeta)$ definierte Sprache nicht. Es gilt also

$$L\left(Gr^2(\mathscr{A}, \mathscr{E}, \zeta)\right) = L\left(Gr^2(\mathscr{A}, \mathscr{E}, \zeta')\right),$$

was zu zeigen war.

Lemma 21. Ist

$$B = L\left(Gr^2(\mathscr{A}, \mathscr{E}, \zeta)\right)$$

und ist ζ stabil, dann ist

$$F(B) = \bigcup_{\substack{i,t,z\\ i>0}} C(Q^{t,z}_{2,i}).$$

Beweis. $F(B)$ ist die Menge der Elemente $z \in Z$, die als erstes Zeichen eines Wortes $w \in B$ auftreten können. Da unsere Sprache links linear ist und da wegen ζ stabil, außer eventuell $(s, 1)$ nur terminale Produktionen $y \to t$ mit $t \in T$ existieren, gilt sicher

$$F(B) \subset \bigcup_{\substack{i,t,z \\ i>0}} C(Q_{2,i}^{t,z})$$

Wir zeigen nun, daß die Inklusion auch in der umgekehrten Richtung gilt.

Sei $x' = x'(i, t, z)$ der Eingang von $\mathscr{A}(i, t, z)$ und $y = y(i, t, z)$ sei ein Ausgang von $\mathscr{A}(i, t, z)$. $G^{t,z}$ ist reduziert, daraus folgt, daß in $G(i, t, z, \zeta)$ die Variable y von x aus zu erreichen ist. Wegen des Zusammenhangsaxioms für $\mathscr{E}$ ist damit aber y in $Gr^j(\mathscr{A}, \mathscr{E}, \zeta)$ sogar von s aus erreichbar. Also kann jede terminale Produktion aus $Q_j^2(y)$ bei der Erzeugung von B auch angewendet werden. Die terminalen Produktionen erzeugen aber nur Anfangszeichen von Wörtern aus B, da Gr eine linkslineare Sprache ist. Also gilt auch

$$\bigcup_{i>0,t,z} C(Q_{2,i}^{t,z}) \subset F(B).$$

Aus beiden Inklusionen folgt die Behauptung des Satzes.

Konstruktion eines Akzeptors für $L(G)$

Wir definieren einen Akzeptor für $L(G)$ mit der Zustandsmenge $\mathfrak{z} = \{(\mathscr{E}, \zeta) \mid \mathscr{E}$ zulässig für $\mathscr{A}, \zeta$ stabiler Zustand von $(\mathscr{A}, \mathscr{E})\}$.

Es sei

$$\Delta: T \times \mathfrak{z} \to \mathfrak{z}$$

die Überführungsfunktion des Akzeptors.

Wie definieren zunächst

$$\nu(\zeta) = \delta(t_1, 1, \zeta)$$

für irgend ein $t_1 \in T$. Das Ergebnis hängt nicht von t_1 ab. Nun definieren wir

$$\mathscr{E}' = \mu(\mathscr{E}, \nu(\zeta), t)$$

durch die Bedingung

$(a, b) \in \mathscr{E}'$ genau dann, wenn $(a, b) \in \mathscr{E}$ ist oder (1) und (2) gelten.

(1) Ist $a = y(j, t_0, z)$ und $b = x'(i, t_1, x)$, dann ist $(y(j, t_0, z), z(x)) \in Q_{1,j}^{t_0,z}$ und $E \cup \{(a, b)\}$ ist zulässig.

(2) Bei gleicher Bezeichnung wie unter (1) gilt $i = H(\mathscr{A}, \mathscr{E}) + 1$ und $t_1 = t$.

$\mathscr{E}'$ entsteht also aus $\mathscr{E}$ durch Hinzunahme weiterer Verbindungsdrähte. Nämlich wir verbinden die im Zustand $\nu(\zeta)$ „aktivierten" Ausgänge y mit zulässigen Eingängen von Automaten $\mathscr{A}(i, t, z)$, die in der nächsten freien Zeile von $\mathscr{A}$ stehen. Hierbei können mehrere Verbindungen auf den gleichen Eingang geschaltet werden, und es können auch mehrere Verbindungen vom gleichen Ausgang ausgehen. Nun bezeichne δ' die Überführungsfunktion von $(\mathscr{A}, \mathscr{E}')$ und ϱ' den in $(\mathscr{A}, \mathscr{E}')$ zu (t, ζ) gehörigen stabilen Zustand.

Damit können wir nun endlich Δ definieren.

$$\Delta(t, \mathscr{E}, \zeta) = \big(\mu(\mathscr{E}, \nu(\zeta), t), \varrho'(t, \nu(\zeta))\big).$$

Wir definieren die Ausgabefunktion Λ durch

$$\Lambda(\mathscr{E}, \zeta) = \eta(s),$$

wo $\eta(s)$ zu ζ gehöriges Ausgangssignal von $(\mathscr{A}, \mathscr{E})$ ist.

Definition. $w \in T^*$ wird genau dann durch

$$\mathfrak{A}(G) = (T, \{0, 1\}, \Lambda, \Delta, \zeta)$$

akzeptiert, wenn bei natürlicher Fortsetzung von Δ auf $T^* \times \mathfrak{z}$ gilt

$$\Lambda(w, \mathscr{E}_0, \zeta_0) = 1.$$

Hierin ist $\mathscr{E}_0 = \emptyset$.

Satz 3. Es ist genau dann $w \in L(G)$, wenn $\mathfrak{A}(G)$ das Wort w akzeptiert.

Wir beweisen zunächst das

Lemma 22. Sei ζ stabiler Zustand von $(\mathscr{A}, \mathscr{E})$ und $\mathscr{E}' = \mu(\mathscr{E}, \nu(\zeta), t)$. Ist $B = L(Gr^2(\mathscr{A}, \mathscr{E}, \zeta), B' = L(Gr^2(\mathscr{A}, \mathscr{E}', \delta'(t, 0, \nu(\zeta))))$, $H(\mathscr{A}, \mathscr{E}) > 0$, dann ist

$$B' = \bigcup_{z \in F(B)} B^t(z) B z.$$

Beweis. Sei $H(\mathscr{A}, \mathscr{E}) = N > 1$ und $\zeta' = \delta'(t, 0, \sigma(\zeta))$. Es ist dann

$$Q_{2,i}^{r,x}(\zeta') = 0 \qquad \text{für } i \leqq N$$

und

$$Q_{2,N+1}^{t,z}(\zeta') = Q_{N+1}^{t,z}$$

falls es $(a, b) \in \mathscr{E}'$ mit $b = z'(N+1, t, z)$ gibt. Dies ist aber genau dann der Fall, wenn ein $Q_{2,j}^{r,x}(\zeta)$ eine Produktion $y_1 \to z$ besitzt.

Nun enthält $\mathscr{S}(\mathscr{E}')$ die Produktion $y(z) \to y_{N+1}^{t,z}$.

Also erhält man nach Lemma 21 B' aus B, indem man in $z' w \in B$ das z durch die aus $y_{N+1}^{t,z}(z)$ erzeugte Sprache ersetzt. Diese Sprache ist aber gerade $B^t(z)$. Tun wir dies für jedes solches z, dann erhalten wir gerade die Behauptung unseres Satzes.

Lemma 23. Sei $\mathscr{E} = \emptyset$ und ζ_0 der Anfangszustand.

Ist

$$(\mathscr{E}', \zeta') = \Delta(t, \mathscr{E}, \zeta),$$

dann ist

$$L(Gr^2(\mathscr{A}, \mathscr{E}', \zeta')) = B^t(s).$$

Beweis. Es ist $\zeta_0(0, t_0, s) = \sigma_0$. Also ist in diesem Zustand $\eta(s) = 1$. s wird nun mit $y_1^{t,z}$ verbunden und im nächsten Schritt wird der Inhalt des Q_2-Registers von $\mathscr{A}(1, t, z)$ auf $Q_1^{t,z}$ gesetzt. Nach spätestens einem weiteren Rechenschritt in $(\mathscr{A}, \mathscr{E}')$ ist der Zustand stabil geworden, d.h. ζ' erreicht. Also gilt in der Tat die Behauptung des Satzes.

Nun können wir unseren Satz beweisen.

Für $l(w) = 0$ ist die Behauptung aufgrund der Konstruktion von $a(S)$ klar.

Der Fall $l(w)=1$ folgt aus Lemma 23.

Wir nehmen an, daß der Satz für $l(w)=k\geqq 1$ bewiesen ist und leiten daraus die Richtigkeit für $k+1$ ab.

Sei also

$$(\mathscr{E},\zeta)=\Delta(w,\emptyset,\zeta_0).$$

Aufgrund der Induktionsvoraussetzung haben wir für $w=tv$, $(\mathscr{E}',\zeta')=\Delta(v,\emptyset,\zeta_0)$

$$(\mathscr{E},\zeta)=\Delta\big(t,\Delta(v,\emptyset,\zeta_0)\big),\qquad B=L\big(Gr^2(\mathscr{A},\mathscr{E}',\zeta')\big)$$

nach Lemma 23.

$$L\big(Gr^2(\mathscr{A},\mathscr{E},\zeta)=\bigcup_{z\in F(B^v(s))} B^t(z)\,B^v_z(s).$$

Hieraus folgt nach Satz 3

$$L\big(Gr^2(\mathscr{A},\mathscr{E},\zeta)\big)=B^w(s)$$

und nach Satz 2 folgt nun

$$w\in L(G)\Leftrightarrow\Delta(\mathscr{A},\mathscr{E},\zeta)=1,$$

was zu zeigen war.

Simulation des Akzeptors $\mathscr{A}$ auf einer RAM

Auf einer Random Access Machine mit Befehlen für Adressensubstitution kann man den Signalfluß durch einen orientierten zykelfreien Graphen in $c\cdot K$ Schritten simulieren, wenn K die Anzahl der Kanten des Graphen ist.

Unser Akzeptor arbeitet in zwei Phasen:

In der ersten Phase vergrößert er den zu $B^u(s)$ gehörigen endlichen Automaten zu dem zu $B^{ut}(s)$ gehörigen endlichen Automaten. Hierbei werden maximal $c_1\cdot 1(n)$ Kanten benötigt. Diese Phase kann also in $c'\cdot 1(n)$ Rechenschritten auf einer RAM simuliert werden.

In der zweiten Phase werden Signale nur „nach unten" weitergegeben, das heißt, es liegt eine Signalausbreitung in einem zykelfreien orientierten Graphen vor. Da dieser Graph maximal $c_2\cdot l^2(u)$ Kanten besitzt, kann dieser Prozeß in $c''\cdot l^2(u)$ Schritten simuliert werden.

Das heißt, daß zwischen dem Lesen von t und r in dem Wort $w=utrv$ maximal $c_0\cdot l^2(u)$ Rechenschritte auf der RAM erforderlich sind. Also benötigt der Akzeptor maximal $c\cdot l^3(w)$ Rechenschritte für die Entscheidung $w\in L$. Damit haben wir den

Satz 4. Das Wortproblem kann für kontextfreie Sprachen durch einen sequentiellen Algorithmus auf einer RAM in cn^3-Schritten entschieden werden.

Literatur

1. Chomsky, N., Schützenberger, M. P.: The algebraic theory of context-free languages. In: Braffort and Hirschberg, Computer Programming and Formal Systems, pp. 118—161. Amsterdam: North Holland 1963
2. Early, J.: An efficient context-free passing algorithm. Comm. ACM **13**, 94—103 (1970)
3. Hotz, G.: Eindeutigkeit und Mehrdeutigkeit formaler Sprachen. EIK **2**, 235—246 (1966)

4. Hotz, G., Claus, V.: Automatentheorie und Formale Sprachen. BI-Hochschulskripten. Mannheim: Bibliographisches Institut 1971
5. Torii, K., Kasami, T., Ozaki, H.: On the complexity of recognition algorithm of context-free languages. J. Inst. Electr. and Comm Engrs. Japan, 264—270 (1967)
6. Rabin, M., Scott, D.: Finite automata and their decision problems. IBM J. R and D **3**, 114—125 (1959)
7. Younger, D. H.: Recognition and parsing of context-free languages in time n^3. Information and Control **10**, 189—208 (1967)

G. Hotz
Fachbereich Angewandte Mathematik
und Informatik
der Universität des Saarlandes
D-6600 Saarbrücken
Bundesrepublik Deutschland