ÜBER DEN MITTLEREN ZEITBEDARF
BEI DER LR(k) - ANALYSE KONTEXTFREIER SPRACHEN

von Herbert Kopp, Saarbrücken

I. EINLEITUNG

Ist G eine Chomskygrammatik, welche eine formale Sprache L definiert, dann besteht das Analyseproblem für G in der Aufgabe, zu gegebenen Worten w zu entscheiden, ob w ein Element der Sprache L ist und in diesem Fall die Ableitungen für w bezüglich G zu finden.

Ein einfaches Verfahren zur Lösung des Analyseproblems für die Klasse der kontextfreien Sprachen ist der LR(k) - Algorithmus von Knuth [3]. Für eine Unter-
menge der kontextfreien Sprachen, nämlich für die Klasse der LR(k) - Sprachen, läßt sich damit die Analyse sogar in Linearzeit durchführen. Im allgemeinen können aber während der Analyse Sackgassen auftreten, so daß die Analyse im schlimmsten Fall sogar exponentiellen Zeitaufwand erfordert.

Um die Brauchbarkeit des LR(k) - Verfahrens für eine gegebene kontextfreie Grammatik beurteilen zu können, sind Antworten auf die folgenden Fragen interessant:

- Wie häufig treten Sackgassen bei der Analyse eines Wortes w durchschnittlich auf?
- Welche Analysezeit benötigt der LR(k) - Algorithmus im Mittel für ein Wort?

Es soll hier über eine Methode berichtet werden, mit der man die mittlere Analysezeit und die Wahrscheinlichkeit für das Auftreten von Sackgassen bei Vorgabe einer kontextfreien Grammatik abschätzen kann. Der Einfachheit halber beschränken wir uns auf die LR(0) - Analyse. Die Erweiterung auf den Fall $k > 0$ bereitet keine Schwierigkeiten.

Mit Hilfe dieser Abschätzungen kann man zeigen, daß der LR(k) - Algorithmus auch für Grammatiken brauchbar ist, die nicht vom Typ LR(k) sind.

II. PRÄZISIERUNG DER PROBLEMSTELLUNG UND LÖSUNGSANSATZ

Wir betrachten eine Chomsky-reduzierte Grammatik $G = \langle \Sigma, \Phi, P, \alpha \rangle$, wobei Σ das nichtterminale Alphabet, Φ das terminale Alphabet, P das Produktionensystem und α das Axiom ist.

Ist $f = x \rightarrow y$ eine Produktionsregel aus $\mathcal{P}$, dann bezeichnen wir mit $Q(f) = x$ und $Z(f) = y$ die linke bzw. die rechte Seite der Produktion f . Weiter betrachten wir zu $f \in \mathcal{P}$ die Menge

$$R(f) = \left\{ wq \left| \begin{array}{l} w \in (\Sigma \setminus \mathcal{V})^* , Z(f) = q , \\ \text{und es gibt eine kanonische} \\ \text{Reduktion } wqv \leftarrow wQ(f)v \leftarrow^* \alpha \end{array} \right. \right\}$$

Zum Begriff der kanonischen Reduktion vergleiche man etwa [1]. Ist $wq \in R(f)$ und $q = Z(f)$ und $v \in \mathcal{V}^*$, dann nennt man das Teilwort q ein Handle des Wortes wqv .

Beispiel: Wir betrachten die Grammatik $\mathcal{G}_1 = \langle \{Y\}, \{a, b\}, \{f_1, f_2\}, Y \rangle$.

mit den Produktionen $f_1 = Y \rightarrow aY$

$f_2 = Y \rightarrow b$

$\mathcal{G}_1$ erzeugt die Sprache $L_1 = \{a^n b \mid n \geq 0\}$

Außerdem gilt $R(f_1) = \{a^n Y \mid n \geq 1\}$

$R(f_2) = \{a^n b \mid n \geq 0\}$

Die Mengen $R(f)$ sind also gerade diejenigen Anfangswörter, die man bei der $LR(0)$ -Analyse gelesen haben muß, um ein Handle zu entdecken, das mit f reduziert werden kann.

Man kann nun zeigen, daß die Mengen $R(f)$ für kontextfreie Grammatiken stets reguläre Sprachen sind, und darüber hinaus lassen sich $LR(0)$ -Grammatiken damit folgendermaßen charakterisieren:

Satz: Eine kontextfreie Chomsky-reduzierte Grammatik ist genau dann vom Typ $LR(0)$, wenn für alle Produktionen $f, g \in \mathcal{P}$ gilt:

$$R(f) \cdot \mathcal{V}^* \cap R(g) = \emptyset \quad \text{falls } f \neq g \text{ ist}$$

$$R(f) \cdot \mathcal{V}^+ \cap R(g) = \emptyset \quad \text{falls } f = g \text{ ist}$$

$LR(0)$ -Grammatiken werden durch diesen Satz also gerade dadurch charakterisiert, daß bei jedem Reduktionsschritt des $LR(0)$ -Verfahrens höchstens ein Handle gefunden werden kann.

Ist nun $\mathcal{G}$ nicht vom Typ $LR(0)$, dann kann es vorkommen, daß im Verlauf der Analyse ein Wort mit mehr als einem Handle auftritt. Es ist dann w von der Form $w = w_1 w_2$ mit $w_2 \in \mathcal{V}^*$ und :

$$w_1 \in M(\mathcal{P}) := \bigcup_{\substack{f, g \in \mathcal{P} \\ f \neq g}} [R(f) \cdot \mathcal{F}^* \cap R(g)] \cup \bigcup_f [R(f) \cdot \mathcal{F}^+ \cap R(f)]$$

In diesem Fall kann die Analyse in Sackgassen geraten, dann nämlich, wenn von den in Frage kommenden Handles ein falsches gewählt wird.

Beispiel: Wir betrachten die Grammatik $\mathcal{G}_2 = \langle \{X, Y\}, \{a, b, c\}, \{f_1, \dots, f_4\}, X \rangle$ mit den Produktionen

$$\begin{aligned} f_1 &= X \rightarrow aY \\ f_2 &= X \rightarrow cc \\ f_3 &= Y \rightarrow Xb \\ f_4 &= Y \rightarrow aY \end{aligned}$$

$\mathcal{G}_2$ erzeugt die Sprache $L_2 = \{a^n ccb^m \mid n \geq m \geq 0\}$. Weiter ist

$$\begin{aligned} R(f_1) &= \{a^n Y \mid n \geq 1\} \\ R(f_2) &= \{a^n cc \mid n \geq 1\} \\ R(f_3) &= \{a^n Xb \mid n \geq 1\} \\ R(f_4) &= \{a^n Y \mid n \geq 2\} \end{aligned}$$

und daraus erhalten wir wie oben angegeben die Menge

$$M(\mathcal{P}) = \{a^n Y \mid n \geq 2\}$$

In $M(\mathcal{P})$ liegen also gerade diejenigen Anfangswörter, die bei der Analyse Konflikte verursachen, da sie sich sowohl mit f_1 als auch mit f_2 reduzieren lassen.

Es läßt sich nun vermuten, daß ein Zusammenhang besteht zwischen der Häufigkeit, mit der Sackgassen auftreten, und damit auch der durchschnittlichen Analysezeit und der "Größe" der Menge $M(\mathcal{P})$.

III. RESULTATE

Bezeichnungen: Ist L eine formale Sprache, dann definieren wir:

$$[L]_r := \left| \left\{ w \in L \mid |w| = r \right\} \right|$$

Weiter sei $\mathcal{G}$ eine kontextfreie Grammatik wie in II. mit $|Z(f)| = 2$ für alle Produktionen $f \in \mathcal{P}$. Diese Einschränkung dient nur zur Vereinfachung der fol -

genden Abschätzungen und hat keine grundsätzliche Bedeutung. Mit T bezeichnen wir die Anzahl der Terminalzeichen von $\mathcal{G}$ und mit k die maximale Anzahl von Produktionen mit gleicher Quelle, d.h.

$$k := \max_{f \in \mathcal{P}} \left| \left\{ g \in \mathcal{P} \mid Q(g) = Q(f) \right\} \right|$$

Damit kann man die folgenden Aussagen zeigen:

Theorem 1 : Für die Wahrscheinlichkeit $q(n)$, daß bei der $LR(0)$ -Analyse eines Wortes $w \in \mathcal{V}^*$ mit $|w| = n$ keine Sackgassen auftreten, gilt:

$$q(n) > 1 - \frac{1}{T^n} \cdot \sum_{i=2}^n [M(\mathcal{P}) \cdot \mathcal{V}^*]_i \cdot k^{n-i}$$

Theorem 2 : Die mittlere Analysezeit für ein Wort $w \in \mathcal{V}^*$ mit $|w| = n$ läßt sich folgendermaßen abschätzen :

1. falls $k < T$ ist :

$$\overline{t(n)} < \frac{n \cdot T \cdot |\mathcal{P}|}{T - k} \cdot \max \left\{ \frac{[R(f)]_r}{T^r} \mid r \leq n \right\}$$

2. falls $k = T$ ist :

$$\overline{t(n)} < \frac{1}{2} \cdot |\mathcal{P}| \cdot (n^2 - 3n + 3) \cdot \max \left\{ \frac{[R(f)]_r}{T^r} \mid r \leq n \right\}$$

Korollar : Für metalinguare Grammatiken mit $k < T$ beträgt die mittlere Analysezeit höchstens

$$\overline{t(n)} < \frac{2 \cdot |\mathcal{N}| \cdot |\mathcal{P}| \cdot (n-1)}{T - k}$$

Wir skizzieren nun noch kurz den Beweis von Theorem 2 :

Eine R -Folge der Ordnung n zur Grammatik $\mathcal{G}$ ist eine Folge $\mathcal{F} = (w_n, \dots, w_r)$, wobei $w_i \in [(\mathcal{N} \cup \mathcal{V}^*)]_i$ ist für $i = r, r+1, \dots, n-1$ und $w_n \in \mathcal{V}^*$ und außerdem $w_n \leftarrow w_{n-1} \leftarrow \dots \leftarrow w_r$ eine kanonische Reduktionsfolge, in der w_r nicht weiter reduzierbar ist.

Wir sagen, w komme auf der m -ten Ebene in der R -Folge $\mathcal{F}$ vor, wenn $w = w_m$ ist für einen Index m mit $r \leq m \leq n$.

Wir untersuchen nun zunächst das folgende Teilproblem: Gegeben sei eine Grammatik G wie oben und gesucht ist die Anzahl der Reduktionsschritte, die ein $LR(0)$ -Automat durchführen muß, um alle R -Folgen der Ordnung n zu erzeugen.

Dazu komme w in einer R -Folge der Ordnung n auf der m -ten Ebene vor. Man kann dann w genau dann mit der Produktion $f \in \mathcal{P}$ reduzieren, wenn $w \in R(f) \cdot \Sigma^*$ ist. Eine solche Reduktion ist eventuell auf mehrere Arten möglich und zwar auf

$$\sum_{r=2}^m \left| \left\{ R(f) \cap (\Sigma^*)^r \right\} \cdot \Sigma^* \cap \{w\} \right|$$

verschiedene Weisen. Variiert man nun noch über alle R -Folgen der Ordnung n , dann kann man auf der m -ten Ebene die Produktion f auf höchstens

$$\sum_{r=2}^m [R(f)]_r \cdot T^{m-r}$$

verschiedene Weisen anwenden. Jede dieser Reduktionsmöglichkeiten kann mehrfach angewandt werden, d.h. bei der Erzeugung verschiedener R -Folgen. Um abzuschätzen, wie oft dies möglich ist, ordnen wir den R -Folgen der Ordnung n einen Graphen zu:

Seine Knoten seien die Wörter $w \in (\Sigma^*)^*$, die in irgendeiner dieser R -Folgen vorkommen.

Der Graph besitzt genau dann eine Kante $u \rightarrow v$, wenn es eine R -Folge der Ordnung n gibt, in der v auf der m -ten und u auf der $(m+1)$ -ten Ebene vorkommen. Da die R -Folgen kanonische Reduktionsfolgen sind, ist klar, daß höchstens k Kanten auf ein und denselben Knoten gerichtet sein können.

Daraus ergibt sich, daß der $LR(0)$ -Algorithmus jede Anwendungsmöglichkeit der Produktion f auf der m -ten Ebene höchstens k^{n-m} mal benutzen kann. Es gibt daher höchstens

$$\sum_{r=2}^m [R(f)]_r \cdot T^{m-r} \cdot k^{n-m}$$

Möglichkeiten, ein Wort auf der m -ten Ebene einer R -Folge mit der Produktion f zu reduzieren.

Summiert man nun noch über alle möglichen Ebenen in den R -Folgen der Ordnung n , dann erhält man eine obere Schranke für die Anzahl $S(n)$ der insgesamt auszuführenden Reduktionsschritte:

$$S(n) \leq \sum_{f \in \mathcal{P}} \sum_{m=2}^n \sum_{r=2}^m [R(f)]_r \cdot T^{m-r} \cdot k^{n-m}$$

Dies kann man leicht umformen zu

$$S(n) \leq \frac{1}{T-k} \cdot \sum_{f \in \mathcal{P}} \left\{ T^{n+1} \sum_{r=2}^n \frac{[R(f)]_r}{T^r} - k^{n+1} \sum_{r=2}^n \frac{[R(f)]_r}{k^r} \right\}$$

Damit erhalten wir als eine obere Schranke für die mittlere Anzahl von Analyse-schritten bei der Analyse von $w \in \mathcal{V}^n$:

$$\overline{t(n)} = \frac{S(n)}{T^n}$$

woraus die in Theorem 2 angegebenen Abschätzungen unmittelbar folgen.

IV. ANWENDUNGEN

Die in II. angegebene Grammatik $\mathcal{G}_2$ ist nicht eindeutig und somit für kein k vom Typ $LR(k)$. Die Abschätzungen von Theorem 1 und Theorem 2 liefern für die Wahrscheinlichkeit $q(n)$ und die mittlere Analysezeit:

$$q(n) > \frac{8}{9} + \frac{1}{4} \cdot \left(\frac{2}{3}\right)^n$$

$$\overline{t(n)} < 8(n-1)$$

Die folgende Grammatik $\mathcal{G}_3$ beschreibt arithmetische Ausdrücke mit unvollständiger Klammerung und Vorrangregeln zwischen den Operatoren:

$$\mathcal{G}_3 = \langle \{\alpha, \tau, \varphi\}, \{V, (,), +, \times\}, \{f_1, \dots, f_6\}, \alpha \rangle$$

mit den Produktionen $f_1 = \varphi \rightarrow V$

$$f_4 = \tau \rightarrow \tau \times \varphi$$

$$f_2 = \varphi \rightarrow (\alpha)$$

$$f_5 = \alpha \rightarrow \tau$$

$$f_3 = \tau \rightarrow \varphi$$

$$f_6 = \alpha \rightarrow \alpha + \tau$$

$\mathcal{G}_3$ ist vom Typ $LR(1)$ und daher können bei der $LR(0)$ -Analyse Sackgassen auftreten. Wir erhalten für diese Grammatik:

$$q(2n) > 0,986 + 0,09 \cdot \left(\frac{6}{25}\right)^n + 0,06 \cdot \left(\frac{3}{25}\right)^n$$

$$\overline{t(n)} < 2$$

LITERATUR :

- [1] G.Hotz, V.Claus : Automatentheorie und formale Sprachen III.
BI - Hochschulschriften 823/823a

- [2] R.Kemp : LR(k) - Analysatoren.
Dissertation, Saarbrücken 1973

- [3] D.E.Knuth: On the translation of languages from left to right.
Information and Control 8, 607-639 (1965)

- [4] H.Kopp: Über die Effizienz von LR(k)-Analysatoren.
Berichte des Fachbereichs Angewandte Mathematik und
Informatik A 74/13 , Universität des Saarlandes