Dyck - Sprachen sind in Bandkomplexität log n analysierbar

Günter Hotz

Jan Messerschmidt

Abstract :

Dyck Languages may be analysed by a two-way-one-counter automaton

Fachbereich 1o

Universität des Saarlandes

Angewandte Mathematik und Informatik

D 66 Saarbrücken

Einleitung :

Seien $X, Y, \overline{Y}$ endliche Alphabete mit

$$X = Y \cup \overline{Y}, \; Y \cap \overline{Y} = \emptyset,$$

und

$$y \rightarrow \overline{y}$$

sei eine bijektive Abbildung von Y auf $\overline{Y}$.

Wir setzen

$$P_Y = \{ 1 \rightarrow y \cdot \overline{y} \mid y \in Y \}$$

worin 1 die Einheit und " • " das Produkt in dem freien Monoid X^* über X ist.

$$D_Y = \{ w \in X^* \mid 1 \xrightarrow[P_Y]{} w \}$$

ist die Menge der Wörter, die sich aus 1 mittels dem Produktionssystem P_X ableiten lassen. D_Y heißt Dycksprache über Y.

Setzen wir

$$P'_Y = \{ y \cdot \overline{y} \rightarrow 1 \mid y \in Y \}$$

dann gilt bekanntlich

$$D_Y = \{ w \in X^* \mid 1 \xrightarrow[P_Y \cup P'_Y]{} w \}.$$

Das heißt, daß die Reduktion der Dycksprachen sackgassenfrei ist.

Wir betrachten nun die folgende <u>Aufgabe</u> :

Auf einem Band stehe ein Wort $w \in X^*$. Es soll eine Turingmaschine T angegeben werden, die jedes w als zwei - Wege - Band lesen darf, und die für jedes w das Problem $w \in D_Y$? unter Verwendung eines möglichst kurzen Rechenbandes entscheiden soll.

Wir nehmen an, daß dem Rechenband das Alphabet $\{ o,1 \}$ zugrunde liegt.

Bekanntlich benötigt man schon im Falle $\# Y = 1$ ($\# Y$ = Anzahl der Elemente von Y) die Bandlänge log n. Kommen wir also für alle Y mit Bandlänge log n aus, dann ist unsere Aufgabe gelöst.

<u>Die Idee des Verfahrens :</u>

Wir sprechen von den Elementen aus Y als öffnende, von den Elementen aus $\overline{Y}$ als schließende Klammern.

Wir zeigen

1. Zu jeder Klammer von $w \in D_Y$ gibt es genau eine Partnerklammer in w derart, daß durch löschen des Paares w in ein Wort $w' \in D_Y$ übergeht, unabhängig davon, was zwischen dem Klammerpaar steht.
2. w liegt in D_Y, wenn jede Klammer von w in genau einem Klammerpaar von w vorkommt.
3. Die Prüfung des Kriteriums in 2. für $w \in X^*$ erfordert eine Bandlänge $\leqslant \lceil \log n \rceil$, wenn n die Länge von w ist.

Zur Präzisierung definieren wir den Monoidhomomorphismus

$$h : X^* \longrightarrow \mathbb{Z} \quad (= \text{Menge der ganzen Zahlen})$$

indem wir setzen

$$h (x) = \begin{cases} 1 \text{ für } x \in Y, \\ -1 \text{ für } x \in \overline{Y}, \end{cases}$$

Es ist nach Definition $h (1) = o$ und $h (y \overline{y}) = h (y) + h (\overline{y}) = o$.

Also ist $h (D_Y) = o$.

Allerdings ist

$$D_Y \neq h^{-1}(o),$$

wie $h(\overline{y}\, y) = o$ und $\overline{y}\, y \notin D_Y$ zeigt.

Für $\# Y = 1$ gilt bekanntlich

$$D_Y = \{ w \in X^* \mid h(w) = o \text{ und } h(n) \geq o$$
$$\text{für alle Prefixe } n \text{ von } w \}.$$

Das heißt, daß sich $w \in D_Y$ durch die successive Berechnung von

$$h(w_1),\ h(w_1\, w_2),\ \ldots,\ h(w_1, \ldots, w_n)$$

entscheiden läßt, wenn $w = (w_1, \ldots, w_n)$, $w_i \in X$ ist.

Wegen

$$| h(w) | \leq \text{Länge}(w) = n$$

läßt sich diese Berechnung in Randkomplexität log n durchführen.

Wir definieren mittels h eine zweite Funktion

$$z_w(i, j) = h(w_i \ldots w_j).$$

Falls keine Mißverständnisse möglich sind, lassen wir den Index w weg.

<u>Definition :</u>

$$(i, j) \text{ mit } 1 \leq i \leq j \leq n$$

heißt ein <u>Klammerpaar</u> von w, genau dann wenn 1. und 2. gilt

1. $z(i, j) = o$ und $w_j = \overline{w}_i$;
2. $z(i, \ell) > o$ für $i \leq \ell < j$.

<u>*Lemma 1 :*</u>

Ist $w \in X^*$ und $1 \leq i \leq n$, dann gibt es höchstens ein j mit

(i, j) oder (j, i) ist Klammerpaar von w. Ist $\dot{z}$ (i, i) = 1, dann ist j > i, im anderen Fall ist j < i.

<u>Beweis :</u>

Dies ist klar für z (i, i) = 1 d.h. $w_i \in Y$, da z (i, ℓ) höchstens eine kleinste Nullstelle größer gleich i besitzt.

Ist z (i, i) = -1, d.h. $w_i \in \overline{Y}$, dann gibt es nach Definition des Klammerpaares kein j > i, so daß (i, j) Klammerpaar ist. Also kann höchstens das größte j < i mit z (j, i) = o mit i ein Klammerpaar bilden.

Ebenso leicht beweist man durch eine Fallunterscheidung :

<u>*Lemma 2 :*</u>

Ist $w = u \cdot v \in X^*$ und ist $w' = u\, y\, \overline{y}\, v$, dann gilt : Jedes i mit 1 ⩽ i ⩽ Länge (w) kommt genau dann in einem Klammerpaar von w vor, wenn dies für jedes j mit 1 ⩽ j ⩽ Länge (w') bezüglich w' gilt.

<u>*Lemma 3 :*</u>

Kommt jedes i mit 1 ⩽ i ⩽ Länge(w) in einem Klammerpaar von w vor, dann gibt es ein ℓ, 1 ⩽ ℓ ⩽ Länge (w) mit

$$w_\ell \in Y \text{ und } w_{\ell+1} = \overline{w}_\ell .$$

<u>Beweis :</u>

Sei (i, j) ein Klammerpaar von w. Ist j = i + 1, dann sind wir fertig. Sei also j > i + 1.

Wir betrachten $z\ (i+1, \ell)$ für $i+1 \leq \ell < j$.

1. $z\ (i+1, i+1) > o$, da sonst $z\ (i, i+1) = o$, was nach Voraussetzung über j nicht zutrifft.
2. $o = h\ (w_i, \ldots, w_j) = h\ (w_i) + h\ (w_{i+i} \ldots w_{j-1}) + h\ (w_j)$
   $= h\ (w_{i+1} \ldots w_{j-1})$ da $w_j = \overline{w}_i$.
   $= z\ (i+1, j-1)$

Hieraus folgt, daß es eine kleine Nullstelle $p > i$ mit $z\ (i+1, p) = o$ gibt.

Kommt i in einem Klammerpaar vor, dann nach Lemma 1 höchstens in $(i+1, p)$. Da jedes i in einem Klammerpaar vorkommt, ist $(i+1, p)$ ein Klammerpaar.

Durch Iteration des Verfahrens finden wir schließlich ein Klammerpaar $(\ell, \ell+1)$, was zu zeigen war.

<u>*Lemma 4 :*</u>

Ist $w \in X^*$ und kommt jedes i mit $1 \leq i \leq n$ in einem Klammerpaar von w vor, dann gilt $w \in D_Y$.

<u>Beweis :</u>

Nach Lemma 3 enthält w ein innerstes Klammerpaar $(i, i+1)$.

Durch Reduktion

$$w \to (w_1 \ldots w_{i-1}, w_{i+2} \ldots w_n)$$

erhalten wir nach Lemma 2 ein Wort w', das ebenfalls die Voraussetzung unseres Lemmas erfüllt. Nach endlich vielen solchen Schritten haben wir w zu 1 reduziert. Also liegt w in D_Y.

Lemma 5 :

Ist $w \in D_Y$, *dann kommt jedes i in einem Klammerpaar vor.*

Beweis :

Die Behauptung ist für $w = 1$ erfüllt und nach Lemma 2 für jedes w, das sich aus 1 ableiten läßt. Also gilt die Behauptung für alle $w \in D_Y$.

Aus Lemma 4 und Lemma 5 folgt :

Satz 1 :

$w \in D_Y$ genau dann, wenn jedes i $(1 \leqslant i \leqslant$ Länge $(w))$ in einem Klammerpaar von w vorkommt.

Um $w \in D_Y$ zu entscheiden, genügt es also zu testen, ob jedes i von w in einem Klammerpaar vorkommt.

Definition des Algorithmus E :

Die Stellung des Lesekopfes auf dem Band speichern wir in i. In der Zelle x merken wir uns jeweils ein Element von X. In z legen wir die Werte von $z(i, j)$ ab.

Wenn wir in der Lage sind für jedes i den Test auf ein j mit (i, j) oder (j, i) Klammerpaar von w so auszuführen, daß wir nach dem Test wieder an der Stelle i sind, dann können wir diesen Test vom ersten Zeichen von w beginnend schrittweise um 1 nach rechts fortschreitend für jedes i der Reihe nach durchführen.

Das Testverfahren KP.

Berechne $h (w_i)$.

Ist $h (w_i) > o$, dann merke w_i und berechne der Reihe nach $z (i, i+1), z (i, i+2), \ldots$ bis zur ersten Nullstelle ℓ, oder bis zum Rand des Bandes. Ist keine Nullstelle vorhanden, dann breche die Rechnung ab, da dann $w \notin D_Y$. Im anderen Falle teste $w_\ell = \overline{w}_i$. Falls der Test negativ ist, breche die Rechnung mit $w \notin D_Y$ ab, im anderen Falle suche das Klammerpaar zu ℓ. Dies ist (i, ℓ).

Ist $h (w_i) < o$ dann merke w_i und berechne der Reihe nach $z (i-1, i), z (i-2, i), \ldots$

Teste $z (\ell, i) = o$ und verfahre symmetrisch zum ersten Fall.

Das Verfahren entscheidet für i in jedem Fall die Existenz eines Klammerpaares und es kehrt zur Stelle i zurück.

Bei diesem Verfahren hat man $z (i, j)$ zu berechnen und w_i zu merken. Dies kann geschehen bei Verwendung eines binären Bandes der Länge $\log_2 n$, w_i kann sich die Steuereinheit merken.

Damit haben wir

Satz 2 :

D_Y ist in Bandkomplexität $\log_2 n$ analysierbar.

Korollar 1 :

Ist R rational über X, dann ist auch $R \cap D_Y$ von Bandkomplexität $\log n$.

Beweis :

Man testet $w \varepsilon D_Y$ in Bandkomplexität log n und danach $w \varepsilon R$ in Bandkomplexität c = konstant.

Korollar 2 :

Ist $\phi : X^* \to Z^*$ ein Homomorphismus mit $\phi(X) = Z$ und $\phi(Y) \cap \phi(\overline{Y}) = \emptyset$, dann hat $R \cap \phi(D_Y)$ die Bandkomplexität log n.

Beweis :

Unter der Voraussetzung des Korollars läßt sich $z_w(i, j)$ auf Z^* verpflanzen, indem man für $u \varepsilon Z^*$ definiert

$$z_u(i, j) = h \cdot \phi^{-1}(u_i \ldots u_j),$$

z_u hat die gleichen Eigenschaften wie $z_w(i, j)$ mit $w \varepsilon X^*$.

Hieraus folgt die Behauptung des Satzes.

Korollar 3 :

Ist $\phi : Z^* \to X^*$ ein Homomorphismus, dann ist $\phi^{-1}(D_Y)$ mit Bandkomplexität log n analysierbar.

Man kann den hier angegebenen Akzeptor für D_Y als two-way-one-counter-Automat auffassen, dessen Zählerinhalt durch die Länge des Eingabe-Bandes beschränkt ist. Also haben wir das

Korollar 4 :

Dycksprachen und die Menge der vollständig geklammerten algebraischen Ausdrücke sind two-way-one-counter-languages mit

Länge (w) - beschränktem Zähler.

Aus diesem Korollar folgt, daß algebraische Ausdrücke, auch wenn deren Zusammensetzung durch Typenvorschriften eingeschränkt wird, in log n Bandkomplexität analysierbar sind.

Zum Schluß eine Bemerkung zur Motivation :

1. Verwendet man zur Analyse von D_Y einen p.d.a. (= push-down-Automaten), dann benötigt man ein Band der Länge n. Also beim Übergang vom p.d.a. zur Turingmaschine geht der Speicherbedarf logarithmisch zurück. Könnte man dies für den writing p.d.a. nachweisen, hätte man das Problem deterministisch c.s. = nicht deterministisch c.s. entschieden.

2. Schlußbemerkung

   Beide Autoren fanden unabhängig voneinander, daß D_Y in Bandkomplexität O (log n) analysierbar ist. Die mitgeteilte Lösung ist die des ersten Autors. Ihr Interesse hierfür verdanken die Verfasser einer mündlichen Mitteilung von M. Nivat.

Anschrift : Fachbereich 1o
der Universität des Saarlandes
Angewandte Mathematik und Informatik
D 66 Saarbrücken