

Einführung in die Theoretische Informatik

Johannes Köbler



Institut für Informatik
Humboldt-Universität zu Berlin

WS 2017/18

Kontextsensitive Sprachen

Definition

Sei $G = (V, \Sigma, P, S)$ eine Grammatik.

- ① G heißt vom Typ 3 oder regulär, falls für alle Regeln $u \rightarrow v$ gilt:

$$u \in V \text{ und } v \in \Sigma V \cup \Sigma \cup \{\varepsilon\}.$$

(d.h. alle Regeln haben die Form $A \rightarrow aB$, $A \rightarrow a$ oder $A \rightarrow \varepsilon$)

- ② G heißt vom Typ 2 oder kontextfrei, falls für alle Regeln $u \rightarrow v$ gilt:

$$u \in V. \quad (\text{d.h. alle Regeln haben die Form } A \rightarrow \alpha)$$

- ③ G heißt vom Typ 1 oder kontextsensitiv, falls für alle Regeln $u \rightarrow v$ gilt:

$$|v| \geq |u|. \quad (\text{mit Ausnahme der } \varepsilon\text{-Sonderregel, s. unten})$$

- ④ Jede Grammatik ist automatisch vom Typ 0.

Die ε -Sonderregel

In einer kontextsensitiven Grammatik ist auch die Regel $S \rightarrow \varepsilon$ zulässig. Aber nur, wenn das Startsymbol S nur links vorkommt.

Bemerkung

- Wie wir gesehen haben, ist CFL in CSL enthalten.
- Zudem ist folgende Sprache nicht kontextfrei:

$$L = \{a^n b^n c^n \mid n \geq 1\}.$$

- L kann jedoch von einer kontextsensitiven Grammatik erzeugt werden.
- Daher ist CFL echt in CSL enthalten.

Beispiel

- Betrachte die kontextsensitive Grammatik $G = (V, \Sigma, P, S)$ mit $V = \{S, B\}$, $\Sigma = \{a, b, c\}$ und den Regeln

$$P: S \rightarrow aSBc, abc \quad (1,2) \quad cB \rightarrow Bc \quad (3) \quad bB \rightarrow bb \quad (4)$$

- In G lässt sich beispielsweise das Wort $w = aabbcc$ ableiten:

$$\underline{S} \xRightarrow{(1)} a\underline{S}Bc \xRightarrow{(2)} aabc\underline{B}c \xRightarrow{(3)} aab\underline{B}cc \xRightarrow{(4)} aabbcc$$

- Allgemein gilt für alle $n \geq 1$:

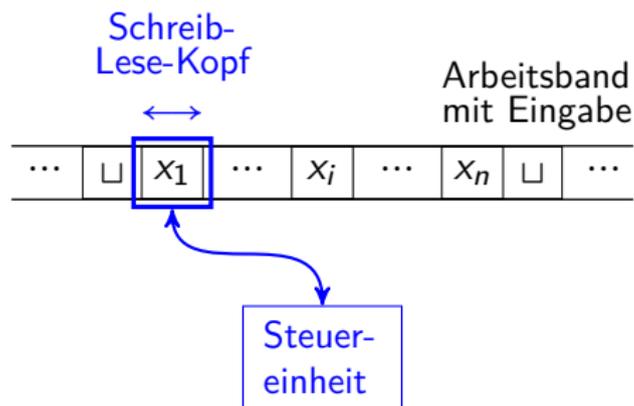
$$\begin{aligned} \underline{S} &\xRightarrow{(1)} a^{n-1} \underline{S} (Bc)^{n-1} \xRightarrow{(2)} a^{n-1} abc \underline{(Bc)^{n-1}} \\ &\xRightarrow{(3)} a^n \underline{bB^{n-1}c^n} \xRightarrow{(4)} a^n b^n c^n \end{aligned}$$

- Also gilt $a^n b^n c^n \in L(G)$ für alle $n \geq 1$.

Beispiel (Schluss)

- Betrachte die kontextsensitive Grammatik $G = (V, \Sigma, P, S)$ mit $V = \{S, B\}$, $\Sigma = \{a, b, c\}$ und den Regeln
$$P: S \rightarrow aSBc, abc \quad (1,2) \quad cB \rightarrow Bc \quad (3) \quad bB \rightarrow bb \quad (4)$$
- Umgekehrt folgt durch Induktion über die Ableitungslänge m , dass jede Satzform $\alpha \in (V \cup \Sigma)^*$ mit $S \Rightarrow^m \alpha$ die folgenden Bedingungen erfüllt:
 - $\#_a(\alpha) = \#_b(\alpha) + \#_B(\alpha) = \#_c(\alpha)$,
 - links von a und links von S kommen nur a 's vor,
 - links von b kommen nur a 's oder b 's vor.
- Daraus ergibt sich, dass in G nur Wörter $w \in \Sigma^*$ der Form $w = a^n b^n c^n$ ableitbar sind, d.h. $L(G) = \{a^n b^n c^n \mid n \geq 1\} \in \text{CSL}$.





- Um ein geeignetes Maschinenmodell für die kontextsensitiven Sprachen zu finden, führen wir zunächst das Rechenmodell der nichtdeterministischen Turingmaschine (NTM) ein.
- Eine NTM erhält ihre Eingabe auf einem nach links und rechts unbegrenzten Band, das in Felder unterteilt ist. Daneben kann sie noch weitere Bänder benutzen, die zu Beginn der Rechnung leer sind.
- Während ihrer Rechnung kann sie den Schreib-Lese-Kopf auf dem Band in beide Richtungen bewegen und dabei die besuchten Bandfelder lesen sowie die gelesenen Zeichen gegebenenfalls überschreiben.

Das Rechenmodell der Turingmaschine

Definition

- Sei $k \geq 1$. Eine **nichtdeterministische k -Band-Turingmaschine** (k -NTM oder einfach **NTM**) wird durch ein 6-Tupel $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ beschrieben, wobei
 - Z eine endliche Menge von Zuständen,
 - Σ das Eingabealphabet (mit $\sqcup \notin \Sigma$; \sqcup heißt Leerzeichen oder Blank),
 - Γ das Arbeitsalphabet (mit $\Sigma \cup \{\sqcup\} \subseteq \Gamma$),
 - $\delta: Z \times \Gamma^k \rightarrow \mathcal{P}(Z \times \Gamma^k \times \{L, R, N\}^k)$ die Überföhrungsfunktion,
 - q_0 der Startzustand und
 - $E \subseteq Z$ die Menge der Endzustände ist.
- Eine k -NTM M heißt **deterministisch** (kurz: M ist eine k -DTM oder einfach **DTM**), falls für alle $(q, a_1, \dots, a_k) \in Z \times \Gamma^k$ gilt:

$$\|\delta(q, a_1, \dots, a_k)\| \leq 1.$$

Das Rechenmodell der Turingmaschine

- Für $(q, b_1, \dots, b_k, D_1, \dots, D_k) \in \delta(p, a_1, \dots, a_k)$ schreiben wir auch $(p, a_1, \dots, a_k) \rightarrow (q, b_1, \dots, b_k, D_1, \dots, D_k)$.
- Eine solche **Anweisung** ist ausführbar, falls
 - p der aktuelle Zustand von M ist und
 - sich für $i = 1, \dots, k$ der Kopf des i -ten Bandes auf einem mit a_i beschrifteten Feld befindet.
- Bei ihrer Ausführung
 - geht M vom Zustand p in den Zustand q über,
 - ersetzt M auf Band i das Symbol a_i durch b_i und
 - bewegt den Kopf auf Band i gemäß D_i (L: ein Feld nach links, R: ein Feld nach rechts, N: keine Bewegung).

Definition

- Eine **Konfiguration** ist ein $(3k + 1)$ -Tupel

$$K = (q, u_1, a_1, v_1, \dots, u_k, a_k, v_k) \in Z \times (\Gamma^* \times \Gamma \times \Gamma^*)^k$$

und besagt, dass

- q der momentane Zustand ist und
 - das i -te Band mit $\dots \sqcup u_i a_i v_i \sqcup \dots$ beschriftet ist, wobei sich der Kopf auf dem Zeichen a_i befindet.
- Im Fall $k = 1$ schreiben wir für eine Konfiguration $K = (q, u, a, v)$ auch kurz $K = uqav$.

Definition

Seien $K = (p, u_1, a_1, v_1, \dots, u_k, a_k, v_k)$ und $K' = (q, u'_1, a'_1, v'_1, \dots, u'_k, a'_k, v'_k)$ Konfigurationen. K' heißt **Folgekonfiguration** von K (kurz $K \vdash K'$), falls eine Anweisung $(p, a_1, \dots, a_k) \rightarrow (q, b_1, \dots, b_k, D_1, \dots, D_k)$ existiert, so dass für $i = 1, \dots, k$ gilt:

im Fall $D_i = N$:	$D_i = R$:	$D_i = L$:
K : $\overline{u_i \boxed{a_i} v_i}$	K : $\overline{u_i \boxed{a_i} v_i}$	K : $\overline{u_i \boxed{a_i} v_i}$
K' : $\overline{u_i \boxed{b_i} v_i}$	K' : $\overline{u_i b_i \boxed{a'_i} v'_i}$	K' : $\overline{u'_i \boxed{a'_i} b_i v_i}$
$u'_i = u_i,$ $a'_i = b_i$ und $v'_i = v_i.$	$u'_i = u_i b_i$ und $a'_i v'_i = \begin{cases} v_i, & v_i \neq \varepsilon, \\ \sqcup, & \text{sonst.} \end{cases}$	$u'_i a'_i = \begin{cases} u_i, & u_i \neq \varepsilon, \\ \sqcup, & \text{sonst} \end{cases}$ und $v'_i = b_i v_i.$

Das Rechenmodell der Turingmaschine

Definition

- Die **Startkonfiguration** von M bei Eingabe $x = x_1 \dots x_n \in \Sigma^*$ ist

$$K_x = \begin{cases} (q_0, \varepsilon, x_1, x_2 \dots x_n, \varepsilon, \sqcup, \varepsilon, \dots, \varepsilon, \sqcup, \varepsilon), & x \neq \varepsilon, \\ (q_0, \varepsilon, \sqcup, \varepsilon, \dots, \varepsilon, \sqcup, \varepsilon), & x = \varepsilon. \end{cases}$$

- Eine **Rechnung** von M bei Eingabe x ist eine (endliche oder unendliche) Folge von Konfigurationen $K_0, K_1, K_2 \dots$ mit $K_0 = K_x$ und $K_0 \vdash K_1 \vdash K_2 \dots$.
- Die von M **akzeptierte** oder **erkannte Sprache** ist

$$L(M) = \{x \in \Sigma^* \mid \exists K \in E \times (\Gamma^* \times \Gamma \times \Gamma^*)^k : K_x \vdash^* K\}.$$

- Ein Wort x wird also genau dann von M akzeptiert (kurz: **$M(x)$ akzeptiert**), wenn es eine Rechnung von M bei Eingabe x gibt, bei der ein Endzustand erreicht wird.

Beispiel

Betrachte die 1-DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ mit $Z = \{q_0, \dots, q_4\}$,
 $\Sigma = \{a, b\}$, $\Gamma = \Sigma \cup \{A, B, \sqcup\}$, $E = \{q_4\}$ und den Anweisungen

$\delta: q_0 a \rightarrow q_1 AR$ (1) Beginn der Schleife: Falls ein a gelesen wird,
ersetze es durch A und ...

$q_1 a \rightarrow q_1 aR$ (2) ... lies a 's und B 's bis ein b kommt (falls kein b

$q_1 B \rightarrow q_1 BR$ (3) kommt, halte ohne zu akzeptieren), ersetze

$q_1 b \rightarrow q_2 BL$ (4) das b durch ein B und ...

$q_2 a \rightarrow q_2 aL$ (5) ... bewege den Kopf wieder nach links bis

$q_2 B \rightarrow q_2 BL$ (6) auf das Feld hinter dem letzten A und

$q_2 A \rightarrow q_0 AR$ (7) gehe zum Beginn der Schleife

$q_0 B \rightarrow q_3 BR$ (8) Falls zu Beginn der Schleife ein B gelesen wird,

$q_3 B \rightarrow q_3 BR$ (9) teste, ob alle Eingabezeichen gelesen wurden

$q_3 \sqcup \rightarrow q_4 \sqcup N$ (10) (wenn ja, dann akzeptiere)

Beispiel (Fortsetzung)

Betrachte die 1-DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ mit $Z = \{q_0, \dots, q_4\}$, $\Sigma = \{a, b\}$, $\Gamma = \Sigma \cup \{A, B, \sqcup\}$, $E = \{q_4\}$ und den Anweisungen

$\delta: q_0a \rightarrow q_1AR$ (1) $q_1a \rightarrow q_1aR$ (2) $q_2a \rightarrow q_2aL$ (5) $q_0B \rightarrow q_3BR$ (8)
 $q_1B \rightarrow q_1BR$ (3) $q_2B \rightarrow q_2BL$ (6) $q_3B \rightarrow q_3BR$ (9)
 $q_1b \rightarrow q_2BL$ (4) $q_2A \rightarrow q_0AR$ (7) $q_3\sqcup \rightarrow q_4\sqcup N$ (10)

- Dann akzeptiert M die Eingabe $aabb$ wie folgt:

$$\begin{array}{cccc}
 q_0aabb \vdash Aq_1abb & \vdash Aaq_1bb & \vdash Aq_2aBb & \vdash q_2AaBb \\
 (1) & (2) & (4) & (5) \\
 \vdash Aq_0aBb & \vdash AAq_1Bb & \vdash AABq_1b & \vdash AAq_2BB \\
 (7) & (1) & (3) & (4) \\
 \vdash Aq_2ABB & \vdash AAq_0BB & \vdash AABq_3B & \vdash AABBq_3\sqcup \\
 (6) & (7) & (8) & (9) \\
 \vdash AABBq_4\sqcup \\
 10
 \end{array}$$

- Ähnlich lässt sich für ein beliebiges $n \geq 1$ zeigen, dass $a^n b^n \in L(M)$ ist.

Beispiel (Schluss)

$\delta: q_0a \rightarrow q_1AR$ (1) $q_1a \rightarrow q_1aR$ (2) $q_2a \rightarrow q_2aL$ (5) $q_0B \rightarrow q_3BR$ (8)
 $q_1B \rightarrow q_1BR$ (3) $q_2B \rightarrow q_2BL$ (6) $q_3B \rightarrow q_3BR$ (9)
 $q_1b \rightarrow q_2BL$ (4) $q_2A \rightarrow q_0AR$ (7) $q_3\sqcup \rightarrow q_4\sqcup N$ (10)

- Andererseits führen die Eingaben aba , abb und aab auf die Rechnungen

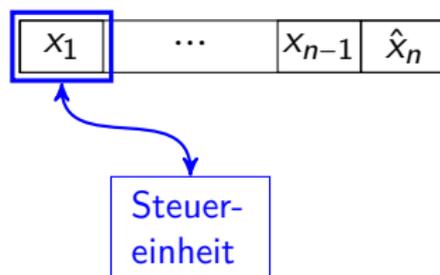
$q_0aba \vdash Aq_1ba \vdash q_2ABa \vdash Aq_0Ba \vdash ABq_3a$ und
 (1) (4) (7) (8)

$q_0abb \vdash Aq_1bb \vdash q_2ABb \vdash Aq_0Bb \vdash ABq_3b$ und
 (1) (4) (7) (8)

$q_0aab \vdash Aq_1ab \vdash Aaq_1b \vdash Aq_2aB \vdash q_2AaB$
 (1) (2) (4) (5)
 $\vdash Aq_0aB \vdash AAq_1B \vdash AABq_1\sqcup$
 (7) (1) (8)

- Da diese nicht fortsetzbar sind und M deterministisch ist, kann M nicht den Endzustand q_4 erreichen, d.h. $aba, abb, aab \notin L(M)$.
- Tatsächlich lässt sich zeigen, dass $L(M) = \{a^n b^n \mid n \geq 1\}$ ist. ◀

- In den Übungen werden wir auch für die Sprache $L' = \{a^n b^n c^n \mid n \geq 1\}$ eine 1-DTM M' konstruieren.
- Wie M besucht auch M' außer den Eingabefeldern nur das erste Blank hinter der Eingabe.
- Dies ist notwendig, damit M' das Ende der Eingabe erkennen kann.
- Falls wir jedoch das letzte Zeichen der Eingabe x markieren, muss der Eingabebereich im Fall $|x| \geq 1$ für diesen Zweck nicht mehr verlassen werden:



- NTMs und DTMs mit dieser Eigenschaft werden auch als LBAs bzw. DLBAs bezeichnet.

Linear beschränkte Automaten

Definition

- Für ein Alphabet Σ sei $\hat{\Sigma} = \Sigma \cup \{\hat{a} \mid a \in \Sigma\}$.
- Für $x = x_1 \dots x_n \in \Sigma^*$ sei $\hat{x} = x_1 \dots x_{n-1} \hat{x}_n$.
- Eine 1-NTM $M = (Z, \hat{\Sigma}, \Gamma, \delta, q_0, E)$ heißt **LBA**, falls gilt:

$$\forall x \in \Sigma^+ : K_{\hat{x}} \vdash^* uqav \Rightarrow |uav| \leq |x|$$
- Die von einem LBA M **akzeptierte** oder **erkannte Sprache** ist

$$L(M) = \{x \in \Sigma^* \mid M(\hat{x}) \text{ akzeptiert}\}$$
- Ein deterministischer LBA wird auch als **DLBA** bezeichnet.
- Die Klasse der **deterministisch kontextsensitiven** Sprachen ist

$$\text{DCSL} = \{L(M) \mid M \text{ ist ein DLBA}\}.$$

Bemerkung

Jede k -NTM, die bei Eingaben der Länge n höchstens linear viele (also $cn + c$ für eine Konstante c) Bandfelder besucht, kann von einem LBA simuliert werden. LBA steht also für **linear beschränkter Automat**.

Beispiel

- Es ist nicht schwer, die 1-DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ mit der Überföhrungsfunktion

$$\begin{aligned} \delta: q_0 a \rightarrow q_1 A R \quad (1) \quad q_1 a \rightarrow q_1 a R \quad (2) \quad q_2 a \rightarrow q_2 a L \quad (5) \quad q_0 B \rightarrow q_3 B R \quad (8) \\ q_1 B \rightarrow q_1 B R \quad (3) \quad q_2 B \rightarrow q_2 B L \quad (6) \quad q_3 B \rightarrow q_3 B R \quad (9) \\ q_1 b \rightarrow q_2 B L \quad (4) \quad q_2 A \rightarrow q_0 A R \quad (7) \quad q_3 \sqcup \rightarrow q_4 \sqcup N \quad (10) \end{aligned}$$

in einen DLBA $M' = (Z, \hat{\Sigma}, \Gamma', \delta', q_0, E)$ für die Sprache $\{a^n b^n \mid n \geq 1\}$ umzuwandeln.

- Ersetze hierzu
 - $\Sigma = \{a, b\}$ durch $\hat{\Sigma} = \{a, b, \hat{a}, \hat{b}\}$ und
 - $\Gamma = \Sigma \cup \{A, B, \sqcup\}$ durch $\Gamma' = \hat{\Sigma} \cup \{A, B, \hat{B}, \sqcup\}$.
- Füge zudem
 - die Anweisungen $q_1 \hat{b} \rightarrow q_2 \hat{B} L$ (4a) und $q_0 \hat{B} \rightarrow q_4 \hat{B} N$ (8a) hinzu und
 - ersetze die Anweisung $q_3 \sqcup \rightarrow q_4 \sqcup N$ (10) durch $q_3 \hat{B} \rightarrow q_4 \hat{B} N$ (10').

Beispiel (Fortsetzung)

- Damit erhalten wir folgende Überföhrungsfunktion für den DLBA M' .

$$\begin{array}{llll}
 \delta': q_0a \rightarrow q_1AR & (1) & q_1\hat{b} \rightarrow q_2\hat{B}L & (4a) & q_0B \rightarrow q_3BR & (8) \\
 q_1a \rightarrow q_1aR & (2) & q_2a \rightarrow q_2aL & (5) & q_0\hat{B} \rightarrow q_4\hat{B}N & (8a) \\
 q_1B \rightarrow q_1BR & (3) & q_2B \rightarrow q_2BL & (6) & q_3B \rightarrow q_3BR & (9) \\
 q_1b \rightarrow q_2BL & (4) & q_2A \rightarrow q_0AR & (7) & q_3\hat{B} \rightarrow q_4\hat{B}N & (10')
 \end{array}$$

- Dieser akzeptiert die beiden Eingaben $a\hat{b}$ und $aab\hat{b}$ wie folgt:

$$\begin{array}{cccccc}
 q_0a\hat{b} & \vdash & Aq_1\hat{b} & \vdash & q_2A\hat{B} & \vdash & Aq_0\hat{B} & \vdash & Aq_4\hat{B} \\
 & & (1) & & (4a) & & (7) & & (8a) \\
 \\
 q_0aab\hat{b} & \vdash & Aq_1abb\hat{b} & \vdash & Aaq_1b\hat{b} & \vdash & Aq_2aB\hat{b} & \vdash & q_2AaB\hat{b} \\
 & & (1) & & (2) & & (4) & & (5) \\
 & \vdash & Aq_0aB\hat{b} & \vdash & AAq_1B\hat{b} & \vdash & AABq_1\hat{b} & \vdash & AAq_2B\hat{b} \\
 & & (7) & & (1) & & (3) & & (4a) \\
 & \vdash & Aq_2AB\hat{b} & \vdash & AAq_0B\hat{b} & \vdash & AABq_3\hat{B} & \vdash & AABq_4\hat{B} \\
 & & (6) & & (7) & & (8) & & (10')
 \end{array}$$



Bemerkung

- Der DLBA M' für die Sprache $\{a^n b^n \mid n \geq 1\}$ aus obigem Beispiel lässt sich leicht in einen DLBA für die kontextsensitive Sprache $\{a^n b^n c^n \mid n \geq 1\}$ transformieren (siehe Übungen).
- Die Sprache $\{a^n b^n c^n \mid n \geq 1\}$ liegt also in $\text{DCSL} \setminus \text{CFL}$.
- Bis heute ungelöst ist die Frage, ob die Klasse DCSL eine echte Teilklasse von CSL ist oder nicht?
- Diese Fragestellung ist als **LBA-Problem** bekannt.

Als nächstes zeigen wir, dass LBAs genau die kontextsensitiven Sprachen erkennen.

Satz

$CSL = \{L(M) \mid M \text{ ist ein LBA}\}.$

Beweis von $CSL \subseteq \{L(M) \mid M \text{ ist ein LBA}\}$

Sei $G = (V, \Sigma, P, S)$ eine kontextsensitive Grammatik. Dann wird $L(G)$ von folgendem LBA M akzeptiert (o.B.d.A. sei $\varepsilon \notin L(G)$):

Arbeitsweise von M bei Eingabe $\hat{x} = x_1 \dots x_{n-1} \hat{x}_n$ mit $n > 0$:

- 1 Markiere das erste Eingabezeichen x_1 mittels \tilde{x}_1 (bzw. \hat{x}_1 mittels $\tilde{\hat{x}}_1$)
- 2 Wähle (nichtdeterministisch) eine Regel $\alpha \rightarrow \beta$ aus P
- 3 Wähle ein beliebiges Vorkommen von β auf dem Band
(falls β nicht vorkommt, halte ohne zu akzeptieren)
- 4 Ersetze die ersten $|\alpha|$ Zeichen von β durch α
- 5 Falls das erste (oder letzte) Zeichen von β markiert war,
markiere auch das erste (letzte) Zeichen von α
- 6 Verschiebe die Zeichen rechts von β um $|\beta| - |\alpha|$ Positionen nach
links und überschreibe die frei werdenden Felder mit Blanks
- 7 Falls das Band abgesehen von Blanks nur das (doppelt markierte)
Startsymbol enthält, halte in einem Endzustand
- 8 Gehe zurück zu Schritt 2

Beweis von $CSL \subseteq \{L(M) \mid M \text{ ist ein LBA}\}$

- Da M sukzessive ein Teilwort β des aktuellen Bandinhalts durch ein Wort α mit $|\alpha| \leq |\beta|$ ersetzt, ist M tatsächlich ein LBA.
- Zudem akzeptiert M eine Eingabe x genau dann, falls es gelingt, eine Ableitung $S \Rightarrow^* x$ in G zu finden (in umgekehrter Reihenfolge von rechts nach links).
- Da sich genau für die Wörter $x \in L(G)$ eine solche Ableitung finden lässt, folgt $L(M) = L(G)$. □

Beweis von $\{L(M) \mid M \text{ ist ein LBA}\} \subseteq \text{CSL}$

- Sei $M = (Z, \hat{\Sigma}, \Gamma, \delta, q_0, E)$ ein LBA (o.B.d.A. sei $\varepsilon \notin L(M)$).
- Betrachte die kontextsensitive Grammatik $G = (V, \Sigma, P, S)$ mit

$$V = \{S, A\} \cup (Z\Gamma \cup \Gamma) \times \Sigma = \{S, A, (q d, a), (d, a) \mid q \in Z, d \in \Gamma, a \in \Sigma\},$$
 die für alle $a, b \in \Sigma$ und $c, c', d \in \Gamma$ folgende Regeln enthält:

$P:$	$S \rightarrow A(\hat{a}, a), (q_0 \hat{a}, a)$	(S)	„Startregeln“
	$A \rightarrow A(a, a), (q_0 a, a)$	(A)	„A-Regeln“
	$(c, a) \rightarrow a$	(F)	„Finale Regeln“
	$(qc, a) \rightarrow a,$	falls $q \in E$	(E) „E-Regeln“
	$(qc, a) \rightarrow (q'c', a),$	falls $qc \rightarrow_M q'c'N$	(N) „N-Regeln“
	$(qc, a)(d, b) \rightarrow (c', a)(q'd, b),$	falls $qc \rightarrow_M q'c'R$	(R) „R-Regeln“
	$(d, a)(qc, b) \rightarrow (q'd, a)(c', b),$	falls $qc \rightarrow_M q'c'L$	(L) „L-Regeln“

Beweis von $\{L(M) \mid M \text{ ist ein LBA}\} \subseteq \text{CSL}$

Beispiel

- Betrachte den LBA $M = (Z, \hat{\Sigma}, \Gamma, \delta, q_0, E)$ mit $Z = \{q_0, \dots, q_4\}$, $\Sigma = \{a, b\}$, $\Gamma = \{a, b, \hat{a}, \hat{b}, A, B, \hat{A}, \hat{B}, \sqcup\}$ und $E = \{q_4\}$, sowie

$$\begin{array}{llll} \delta: q_0 a \rightarrow q_1 A R & q_1 b \rightarrow q_2 B L & q_2 A \rightarrow q_0 A R & q_0 \hat{B} \rightarrow q_4 \hat{B} N \\ q_1 a \rightarrow q_1 a R & q_1 \hat{b} \rightarrow q_2 \hat{B} L & q_2 B \rightarrow q_2 B L & q_3 B \rightarrow q_3 B R \\ q_1 B \rightarrow q_1 B R & q_2 a \rightarrow q_2 a L & q_0 B \rightarrow q_3 B R & q_3 \hat{B} \rightarrow q_4 \hat{B} N \end{array}$$

- Die zugehörige kontextsensitive Grammatik $G = (V, \Sigma, P, S)$ hat dann die Variablenmenge

$$\begin{aligned} V &= \{S, A\} \cup (Z\Gamma \cup \Gamma) \times \Sigma \\ &= \{S, A, (q_i c, a), (q_i c, b), (c, a), (c, b) \mid 0 \leq i \leq 4, c \in \Gamma\} \end{aligned}$$

- Die Regelmenge P von G enthält folgende Start- und A-Regeln:

$$\begin{array}{ll} S \rightarrow A(\hat{a}, a), A(\hat{b}, b), (q_0 \hat{a}, a), (q_0 \hat{b}, b) & (S_1-S_4) \\ A \rightarrow A(a, a), A(b, b), (q_0 a, a), (q_0 b, b) & (A_1-A_4) \end{array}$$

Beweis von $\{L(M) \mid M \text{ ist ein LBA}\} \subseteq \text{CSL}$

Beispiel (Fortsetzung)

- Zudem enthält P wegen $E = \{q_4\}$ für jedes $c \in \Gamma$ die F- und E-Regeln

$$(c, a) \rightarrow a, (c, b) \rightarrow b \quad (F_1-F_{16})$$

$$(q_4c, a) \rightarrow a, (q_4c, b) \rightarrow b \quad (E_1-E_{16})$$

- Daneben enthält P beispielsweise noch folgende Regeln:

- Wegen der Anweisung $q_3\hat{B} \rightarrow q_4\hat{B}N$ die beiden N-Regeln

$$(q_3\hat{B}, a) \rightarrow (q_4\hat{B}, a) \text{ und } (q_3\hat{B}, b) \rightarrow (q_4\hat{B}, b)$$

- Wegen der Anweisung $q_1b \rightarrow q_2BL$ für jedes $d \in \Gamma$ die 4 L-Regeln

$$(d, a)(q_1b, a) \rightarrow (q_2d, a)(B, a) \quad (d, a)(q_1b, b) \rightarrow (q_2d, a)(B, b)$$

$$(d, b)(q_1b, a) \rightarrow (q_2d, b)(B, a) \quad (d, b)(q_1b, b) \rightarrow (q_2d, b)(B, b)$$

- Wegen der Anweisung $q_0a \rightarrow q_1AR$ für jedes $d \in \Gamma$ die 4 R-Regeln

$$(q_0a, a)(d, a) \rightarrow (A, a)(q_1d, a) \quad (q_0a, a)(d, b) \rightarrow (A, a)(q_1d, b)$$

$$(q_0a, b)(d, a) \rightarrow (A, b)(q_1d, a) \quad (q_0a, b)(d, b) \rightarrow (A, b)(q_1d, b) \triangleleft$$

Beweis von $\{L(M) \mid M \text{ ist ein LBA}\} \subseteq \text{CSL}$

- Sei $M = (Z, \hat{\Sigma}, \Gamma, \delta, q_0, E)$ ein LBA (o.B.d.A. sei $\varepsilon \notin L(M)$).
- Betrachte die kontextsensitive Grammatik $G = (V, \Sigma, P, S)$ mit

$$V = \{S, A\} \cup (Z\Gamma \cup \Gamma) \times \Sigma = \{S, A, (q d, a), (d, a) \mid q \in Z, d \in \Gamma, a \in \Sigma\},$$
 die für alle $a, b \in \Sigma$ und $c, c', d \in \Gamma$ folgende Regeln enthält:

$P:$	$S \rightarrow A(\hat{a}, a), (q_0 \hat{a}, a)$	(S)	„Startregeln“
	$A \rightarrow A(a, a), (q_0 a, a)$	(A)	„A-Regeln“
	$(c, a) \rightarrow a$	(F)	„Finale Regeln“
	$(qc, a) \rightarrow a,$	(E)	„E-Regeln“
	falls $q \in E$		
	$(qc, a) \rightarrow (q'c', a),$	(N)	„N-Regeln“
	falls $qc \rightarrow_M q'c'N$		
	$(qc, a)(d, b) \rightarrow (c', a)(q'd, b),$	(R)	„R-Regeln“
	falls $qc \rightarrow_M q'c'R$		
	$(d, a)(qc, b) \rightarrow (q'd, a)(c', b),$	(L)	„L-Regeln“
	falls $qc \rightarrow_M q'c'L$		

Beweis von $\{L(M) \mid M \text{ ist ein LBA}\} \subseteq \text{CSL}$

- Durch Induktion über m lässt sich nun leicht für alle $a_1, \dots, a_n \in \Gamma$ und $q \in Z$ die folgende Äquivalenz beweisen:

$$q_0 x_1 \dots x_{n-1} \hat{x}_n \vdash^m a_1 \dots a_{i-1} q a_i \dots a_n \text{ gdw.}$$

$$(q_0 x_1, x_1) \dots (\hat{x}_n, x_n) \xRightarrow{(N,R,L)}^m (a_1, x_1) \dots (q a_i, x_i) \dots (a_n, x_n)$$

- Ist also $q_0 x_1 \dots x_{n-1} \hat{x}_n \vdash^m a_1 \dots a_{i-1} q a_i \dots a_n$ eine akzeptierende Rechnung von $M(x_1 \dots x_{n-1} \hat{x}_n)$ mit $q \in E$, so folgt

$$\begin{aligned} S &\xRightarrow{(S)} A(\hat{x}_n, x_n) \xRightarrow{(A)}^{n-1} (q_0 x_1, x_1)(x_2, x_2) \dots (x_{n-1}, x_{n-1})(\hat{x}_n, x_n) \\ &\xRightarrow{(N,L,R)}^m (a_1, x_1) \dots (a_{i-1}, x_{i-1})(q a_i, x_i) \dots (a_n, x_n) \xRightarrow{(F,E)}^n x_1 \dots x_n \end{aligned}$$

- Die Inklusion $L(G) \subseteq L(M)$ folgt analog. □

Bemerkung

Eine einfache Modifikation des Beweises zeigt, dass 1-NTMs genau die Sprachen vom Typ 0 akzeptieren (siehe Übungen).

	Vereinigung	Schnitt	Komplement	Produkt	Sternhülle
REG	ja	ja	ja	ja	ja
DCFL	nein	nein	ja	nein	nein
CFL	ja	nein	nein	ja	ja
DCSL	ja	ja	ja	ja	ja
CSL	ja	ja	ja	ja	ja
RE	ja	ja	nein	ja	ja

- In der VL Komplexitätstheorie wird gezeigt, dass die Klasse CSL unter Komplementbildung abgeschlossen ist.
- Im nächsten Kapitel werden wir sehen, dass die Klasse RE nicht unter Komplementbildung abgeschlossen ist.
- Die übrigen Abschlusseigenschaften der Klassen DCSL, CSL und RE in obiger Tabelle werden in den Übungen bewiesen.