

Einführung in die Theoretische Informatik

Johannes Köbler



Institut für Informatik
Humboldt-Universität zu Berlin

WS 2020/21

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:

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

(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:

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

- 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, falls das Startsymbol S nicht auf der rechten Seite einer Regel vorkommt

CFL ist echt in CSL enthalten

Bemerkung

- Wie wir gesehen haben, ist CFL in CSL enthalten
- Zudem ist die Sprache $L = \{a^n b^n c^n \mid n \geq 1\}$ nicht kontextfrei
- L kann jedoch von einer kontextsensitiven Grammatik erzeugt werden (siehe nächste Folie)
- 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

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

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

$$\begin{array}{ccccccc} S & \xrightarrow{(1)} & a \underline{S} B c & \xrightarrow{(2)} & a a \underline{B} c B c & \xrightarrow{(3)} & a a b \underline{B} c c \\ & & & & & & \xrightarrow{(4)} \\ & & & & & & a a b b c c \end{array}$$

- Allgemein gilt für alle $n \geq 1$:

$$\begin{array}{ccccccc} S & \xrightarrow{(1)} & a^{n-1} \underline{S} (Bc)^{n-1} & \xrightarrow{(2)} & a^{n-1} \underline{abc} (Bc)^{n-1} \\ & & & & & & \\ & \xrightarrow{(3)} & a^n b \underline{B^{n-1} c^n} & & \xrightarrow{(4)} & a^n b^n c^n \end{array}$$

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

Beispiel (Fortsetzung)

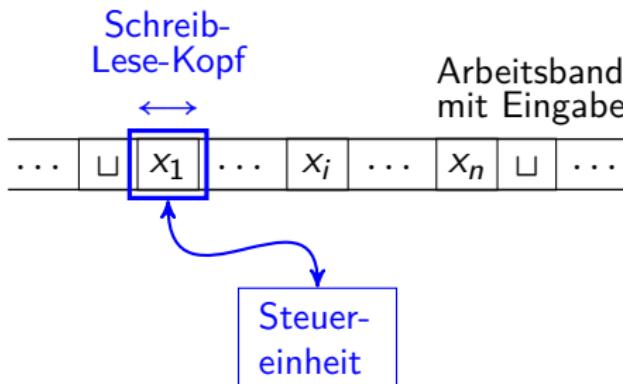
- Betrachte die kontextsensitive Grammatik $G = (V, \Sigma, P, S)$ mit $V = \{S, B\}$, $\Sigma = \{a, b, c\}$ und den Regeln

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

- 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. $L(G) = \{a^n b^n c^n \mid n \geq 1\} \in \text{CSL}$

△

Die Turingmaschine



- 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; zudem kann sie weitere Bänder benutzen, die zu Beginn der Rechnung komplett leer sind
- In jedem Rechenschritt kann sie die aktuell besuchten Bandfelder lesen, die gelesenen Zeichen überschreiben und den Schreib-Lese-Kopf auf jedem Band um maximal ein Feld nach links oder rechts bewegen

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. Dabei ist
 - 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 Überführungsfunktion
 - q_0 der Startzustand und
 - $E \subseteq Z$ die Menge der Endzustände
- 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 auf Band $i = 1, \dots, k$ das Symbol a_i durch b_i und
 - bewegt den Kopf auf Band $i = 1, \dots, k$ gemäß D_i
(L: ein Feld nach links, R: ein Feld nach rechts, N: keine Bewegung)

Das Rechenmodell der Turingmaschine

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

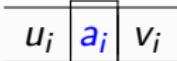
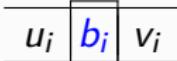
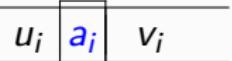
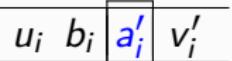
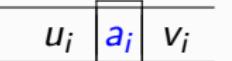
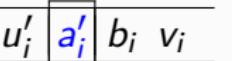
$$K = (q, u_1, a_1, v_1, \dots, u_k, a_k, v_k) \in Z \times (\Gamma^* \times \Gamma \times \Gamma^*)^k$$

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 $k = 1$ notieren wir eine Konfiguration $K = (q, u, a, v)$ auch in der Form $K = uqav$

Das Rechenmodell der Turingmaschine

- 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:

| $D_i = N$ | $D_i = R$ | $D_i = L$ |
|---|---|---|
| $K:$  $K':$  | $K:$  $K':$  | $K:$  $K':$  |
| $u'_i = u_i$ $a'_i = b_i$ $v'_i = v_i$ | $u'_i = u_i b_i$ $a'_i v'_i = \begin{cases} v_i, & v_i \neq \varepsilon \\ \sqcup, & \text{sonst} \end{cases}$ | $u'_i a'_i = \begin{cases} u_i, & u_i \neq \varepsilon \\ \sqcup, & \text{sonst} \end{cases}$ $v'_i = b_i v_i$ |

Das Rechenmodell der Turingmaschine

- Man beachte, dass sich die Länge der Bandinschrift $u_i a_i v_i$ beim Übergang von K zu K' genau dann um 1 erhöht, wenn in K' zum ersten Mal ein neues Feld auf dem i -ten Band besucht wird
- Andernfalls bleibt die Länge von $u_i a_i v_i$ unverändert
- Die Länge von $u_i a_i v_i$ entspricht also genau der Anzahl der bisher auf dem i -ten Band besuchten Felder (inkl. Eingabezeichen im Fall $i = 1$)
- 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) Konfigurationenfolge $K_0, K_1, K_2 \dots$ mit $K_0 = K_x$ und $K_0 \vdash K_1 \vdash K_2 \dots$
- Die von M **akzeptierte** oder **erkannte Sprache** ist

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

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

Das Rechenmodell der Turingmaschine

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

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

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

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

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

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

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

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

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

Das Rechenmodell der Turingmaschine

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

$$\begin{array}{llll}
 \delta: q_0a \rightarrow q_1AR & (1) & q_1a \rightarrow q_1aR & (2) \\
 & & q_2a \rightarrow q_2aL & (5) \\
 & & q_1B \rightarrow q_1BR & (3) \\
 & & q_2B \rightarrow q_2BL & (6) \\
 & & q_3B \rightarrow q_3BR & (9) \\
 q_1b \rightarrow q_2BL & (4) & q_2A \rightarrow q_0AR & (7) \\
 & & q_3\sqcup \rightarrow q_4\sqcup N & (10)
 \end{array}$$

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

$$\begin{array}{llllll}
 q_0aabb & \xleftarrow{(1)} & A\color{blue}{q_1abb} & \xleftarrow{(2)} & Aa\color{blue}{q_1bb} & \xleftarrow{(4)} & A\color{blue}{q_2aBb} \\
 & & \xleftarrow{(7)} & & \xleftarrow{(1)} & & \xleftarrow{(5)} \\
 & & A\color{blue}{q_0aBb} & & AA\color{blue}{q_1Bb} & \xleftarrow{(3)} & AAB\color{blue}{q_1b} \\
 & & \xleftarrow{(6)} & & \xleftarrow{(4)} & & \xleftarrow{(4)} \\
 & & A\color{blue}{q_2ABB} & \xleftarrow{(6)} & AA\color{blue}{q_0BB} & \xleftarrow{(7)} & AAB\color{blue}{q_3B} \\
 & & \xleftarrow{(10)} & & \xleftarrow{(8)} & & \xleftarrow{(9)} \\
 & & A\color{blue}{ABBq_4\sqcup} & & & &
 \end{array}$$

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

Das Rechenmodell der Turingmaschine

Beispiel (Schluss)

$$\begin{array}{llll}
 \delta: q_0a \rightarrow q_1AR & (1) & q_1a \rightarrow q_1aR & (2) \\
 & & q_2a \rightarrow q_2aL & (5) \\
 & & q_1B \rightarrow q_1BR & (3) \\
 & & q_2B \rightarrow q_2BL & (6) \\
 & & q_3B \rightarrow q_3BR & (9) \\
 q_1b \rightarrow q_2BL & (4) & q_2A \rightarrow q_0AR & (7) \\
 & & q_3\sqcup \rightarrow q_4\sqcup N & (10)
 \end{array}$$

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

$$q_0 \textcolor{blue}{a} \textcolor{blue}{b} \textcolor{blue}{a} \vdash A \textcolor{blue}{q_1} \textcolor{blue}{b} \textcolor{blue}{a} \quad (1) \quad \textcolor{blue}{q_1} \textcolor{blue}{b} \textcolor{blue}{a} \vdash \textcolor{blue}{q_2} A \textcolor{blue}{B} \textcolor{blue}{a} \quad (4) \quad \textcolor{blue}{q_2} A \textcolor{blue}{B} \textcolor{blue}{a} \vdash A \textcolor{blue}{q_0} \textcolor{blue}{B} \textcolor{blue}{a} \quad (7) \quad A \textcolor{blue}{q_0} \textcolor{blue}{B} \textcolor{blue}{a} \vdash AB \textcolor{red}{q_3} a \quad \text{und} \quad (8)$$

$$q_0 \textcolor{blue}{a} \textcolor{blue}{b} \textcolor{blue}{b} \vdash A \textcolor{blue}{q_1} \textcolor{blue}{b} \textcolor{blue}{b} \quad (1) \quad \textcolor{blue}{q_1} \textcolor{blue}{b} \textcolor{blue}{b} \vdash \textcolor{blue}{q_2} A \textcolor{blue}{B} \textcolor{blue}{b} \quad (4) \quad \textcolor{blue}{q_2} A \textcolor{blue}{B} \textcolor{blue}{b} \vdash A \textcolor{blue}{q_0} \textcolor{blue}{B} \textcolor{blue}{b} \quad (7) \quad A \textcolor{blue}{q_0} \textcolor{blue}{B} \textcolor{blue}{b} \vdash AB \textcolor{red}{q_3} b \quad \text{und} \quad (8)$$

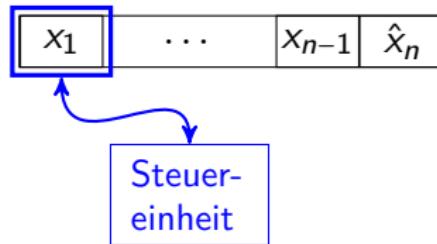
$$\begin{array}{llll}
 q_0 \textcolor{blue}{a} \textcolor{blue}{a} \textcolor{blue}{b} \vdash A \textcolor{blue}{q_1} \textcolor{blue}{a} \textcolor{blue}{b} \quad (1) & \textcolor{blue}{q_1} \textcolor{blue}{a} \textcolor{blue}{b} \vdash A \textcolor{blue}{a} \textcolor{blue}{q_1} \textcolor{blue}{b} \quad (2) & \textcolor{blue}{a} \textcolor{blue}{a} \textcolor{blue}{q_1} \textcolor{blue}{b} \vdash A \textcolor{blue}{q_2} \textcolor{blue}{a} \textcolor{blue}{B} \quad (4) & \textcolor{blue}{q_2} \textcolor{blue}{a} \textcolor{blue}{B} \vdash \textcolor{blue}{q_2} A \textcolor{blue}{a} \textcolor{blue}{B} \quad (5) \\
 & & & \vdash A \textcolor{blue}{q_0} \textcolor{blue}{a} \textcolor{blue}{B} \quad (7) \\
 & & & \vdash A A \textcolor{blue}{q_1} \textcolor{blue}{B} \quad (1) \\
 & & & \vdash A A B \textcolor{red}{q_1} \sqcup \quad (8)
 \end{array}$$

- 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



Ein Maschinenmodell für CSL

- In den Übungen werden wir eine 1-DTM M' für die Sprache $L' = \{a^n b^n c^n \mid n \geq 1\}$ 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 benutzt, kann von einem LBA simuliert werden; LBA steht also für **linear beschränkter Automat**

Linear beschränkte Automaten

Beispiel

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

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

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')

Linear beschränkte Automaten

Beispiel (Fortsetzung)

- Damit erhalten wir folgende Überführungsfunktion für den DLBA M' :

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

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

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

Das LBA-Problem

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

Charakterisierung von CSL mittels LBAs

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

Satz

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

Bemerkung

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

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

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 auf dem Band das (doppelt markierte) Startsymbol erscheint,
halte in einem Endzustand
- 8 Gehe zurück zu Schritt 2

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

- 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
 $V = \{S, A\} \cup (Z\Gamma \cup \Gamma) \times \Sigma = \{S, A, (qd, a), (d, a) \mid q \in Z, d \in \Gamma, a \in \Sigma\}$,
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

$$\begin{array}{llll}
 \delta: & q_0a \rightarrow q_1AR & q_1b \rightarrow q_2BL & q_2A \rightarrow q_0AR \\
 & q_1a \rightarrow q_1aR & q_1\hat{b} \rightarrow q_2\hat{B}L & q_2B \rightarrow q_2BL \\
 & q_1B \rightarrow q_1BR & q_2a \rightarrow q_2aL & q_0B \rightarrow q_3BR \\
 & & & q_3B \rightarrow q_3BR \\
 & & & q_3\hat{B} \rightarrow q_4\hat{B}N
 \end{array}$$

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

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

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

$$\begin{array}{ll}
 S \rightarrow A(\hat{a}, a), A(\hat{b}, b), (q_0\hat{a}, a), (q_0\hat{b}, b) & (S_1-S_4) \\
 A \rightarrow A(a, a), A(b, b), (q_0a, a), (q_0b, b) & (A_1-A_4)
 \end{array}$$

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

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

$$(q_4 c, a) \rightarrow a, (q_4 c, b) \rightarrow b \quad (E_1-E_{16})$$

- Schließlich enthält P noch 4 N-, 128 L- und 192 R-Regeln wie z.B.

- für die Anweisung $q_3 \hat{B} \rightarrow q_4 \hat{B} N$ die beiden folgenden N-Regeln:

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

- für $q_1 b \rightarrow q_2 B L$ insgesamt 32 L-Regeln, nämlich für jedes $d \in \Gamma$:

$$(d, a)(q_1 b, a) \rightarrow (q_2 d, a)(B, a) \quad (d, a)(q_1 b, b) \rightarrow (q_2 d, a)(B, b)$$

$$(d, b)(q_1 b, a) \rightarrow (q_2 d, b)(B, a) \quad (d, b)(q_1 b, b) \rightarrow (q_2 d, b)(B, b)$$

- für $q_0 a \rightarrow q_1 A R$ insgesamt 32 R-Regeln, nämlich für jedes $d \in \Gamma$:

$$(q_0 a, a)(d, a) \rightarrow (A, a)(q_1 d, a) \quad (q_0 a, a)(d, b) \rightarrow (A, a)(q_1 d, b)$$

$$(q_0 a, b)(d, a) \rightarrow (A, b)(q_1 d, a) \quad (q_0 a, b)(d, b) \rightarrow (A, b)(q_1 d, b) \quad \triangleleft$$

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
 $V = \{S, A\} \cup (Z\Gamma \cup \Gamma) \times \Sigma = \{S, A, (qd, a), (d, a) \mid q \in Z, d \in \Gamma, a \in \Sigma\}$,
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 $\{L(M) \mid M \text{ ist ein LBA}\} \subseteq \text{CSL}$

- 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 im Fall $n = 1$

$$S \xrightarrow{(S)} (q_0 \hat{x}_1, x_1) \xrightarrow{(N)} (q a_1, x_1) \xrightarrow{(E)} x_1$$

und im Fall $n \geq 2$

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

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

□

Zusammenfassung der Abschlusseigenschaften

| | 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 zum Teil in den Übungen bewiesen