

Formale Grundlagen der Informatik

11

Pushdown-Automaten (Kellerautomaten)

Recap: Ein einfacher Parser

- Aufbau eines Ableitungsbaums von der Wurzel zum Eingabewort („Top-Down-Parsing“)
- Arbeitet stets am linken Ende der bisher gefundenen Satzform
 - Terminal: Abhaken
 - Nichtterminal: Aktivieren (also als nächstes Ersetzen)

➤ Aufbau einer **Linksableitung**

$$E \Rightarrow_L \text{id} * E \Rightarrow_L \text{id} * \text{id}$$

- **Backtracking** bei Fehlern
- Implementierung mit einem **Stack (linkes Symbol am Top)**

Recap: Ein einfacher Parser

$$E \rightarrow \text{id} + E \mid \text{id} * E \mid \text{id}$$

Stack	Eingabe	Aktion
E	$\text{id} * \text{id}$	$E \rightarrow \text{id} + E$
$\text{id} + E$	$\text{id} * \text{id}$	Terminal
$+ E$	$* \text{id}$	Backtrack
E	$\text{id} * \text{id}$	$E \rightarrow \text{id} * E$
$\text{id} * E$	$\text{id} * \text{id}$	Terminal
$* E$	$* \text{id}$	Terminal
E	id	$E \rightarrow \text{id} + E$
$\text{id} + E$	id	Terminal
$+ E$	ε	Backtrack

Recap: Ein einfacher Parser

$$E \rightarrow \text{id} + E \mid \text{id} * E \mid \text{id}$$

Stack	Eingabe	Aktion
E	id	$E \rightarrow \text{id} * E$
id * E	id	Terminal
* E	ε	Backtrack
E	id	$E \rightarrow \text{id}$
id	id	Terminal
ε	ε	ACCEPT

statt
Backtracking
kann auch
nicht-
deterministisch
die nächste
Regel „geraten“
werden

ACCEPT wenn Stack leer und alle Eingabesymbole abgehakt/gelesen

Pushdown-Automaten – Idee

- arbeiten wie NEA (Nichtdeterminismus statt Backtracking)
- speichern zusätzliche Information in einem **Pushdown**
 - kann (nur) das Top-Symbol lesen
 - ersetzt Top-Symbol durch eine Zeichenkette
 - durch ε , wenn Top-Symbol gelöscht werden soll („pop“)
 - Transition hängt ab von
 1. Zustand
 2. Eingabesymbol
 3. Top-Symbol im Pushdown
- Arbeit bei leerem Keller nicht möglich → initiales Kellersymbol

Pushdown-Automaten – Definition

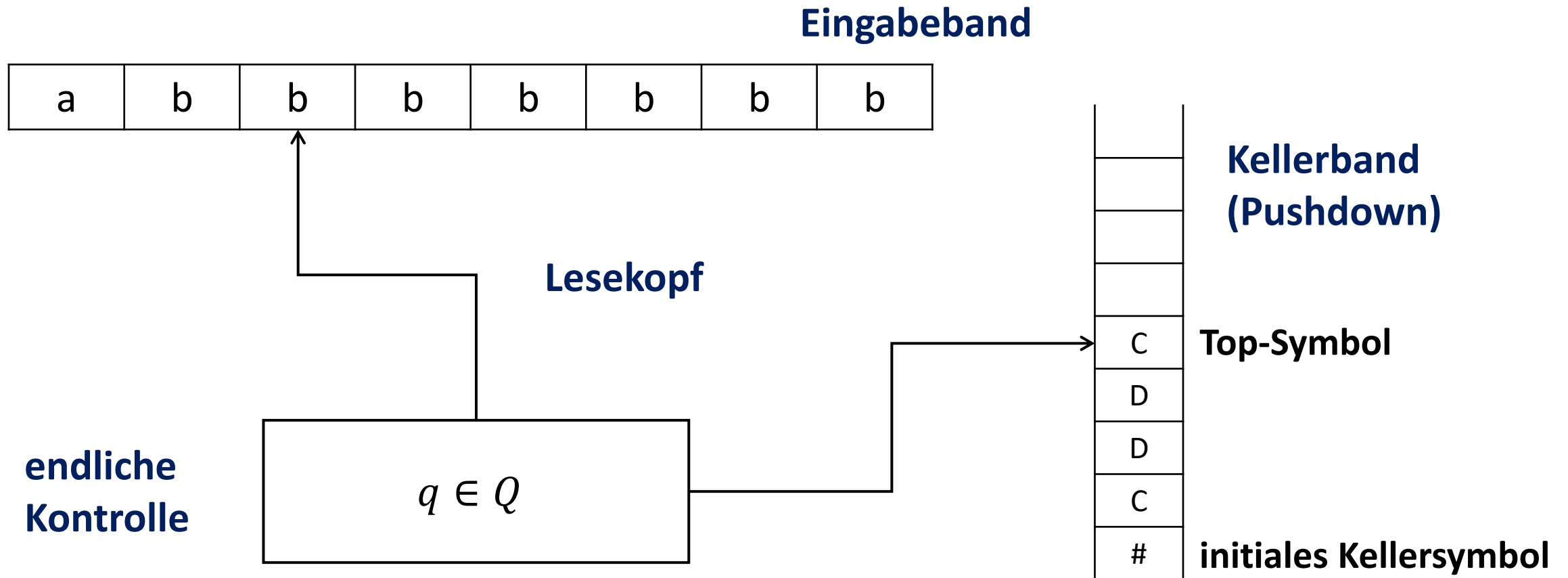
Ein **Pushdown-Automat (PDA) / Kellerautomat** ist ein 7-Tupel

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F).$$

Dabei ist

- Q eine endliche Menge von Zuständen,
- Σ ein Alphabet der Eingabesymbole,
- Γ ein Alphabet der Kellersymbole,
- $q_0 \in Q$ der Startzustand,
- $\# \in \Gamma$ das initiale Kellersymbol,
- $F \subseteq Q$ die Menge der akzeptierenden Zustände,
- $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow 2^{Q \times \Gamma^*}$ die Überföhrungsfunktion.

Pushdown-Automaten – Veranschaulichung



PDA - Zustandsbeschreibungen

- Sei $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$ ein PDA.
- Zustand: Paar aus $Q \times \Gamma$.
- **Konfiguration:** Zustand plus unverbrauchte Eingabe
 - Tripel aus $Q \times \Gamma \times \Sigma^*$
- **Konfigurationsübergang:**

$$(q, aw, Z\alpha) \vdash (p, w, \beta\alpha) \text{ falls } (p, \beta) \in \delta(q, a, Z) \quad (a \in \Sigma \cup \{\varepsilon\})$$
 - β ersetzt das Top-Symbol Z
- \vdash^* ist die reflexive und transitive Hülle von \vdash
- \vdash^i für i aufeinanderfolgende Konfigurationsübergänge

PDA – akzeptierte Sprache

Sei $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$ ein PDA.

Der PDA M akzeptiert eine Sprache ...

1. ... durch akzeptierenden Zustand:

$$L(M) = \{ w \in \Sigma^* \mid (q_0, w, \#) \vdash^* (f, \varepsilon, \gamma) \text{ für ein } f \in F \text{ und ein } \gamma \in \Gamma^* \}$$

2. ... durch leeren Keller:

$$N(M) = \{ w \in \Sigma^* \mid (q_0, w, \#) \vdash^* (p, \varepsilon, \varepsilon) \text{ für ein } p \in Q \}$$

PDA – Beispiel (1)

$$L = \{ ww^R \mid w \in \{0,1\}^* \}$$

$M_1 = (\{q_0, q_1, q_2\}, \{0,1\}, \{0,1, \#\}, \delta_1, q_0, \#, \{q_2\})$ mit

$$\delta_1(q_0, 0, \#) = \{(q_0, 0\#)\}$$

$$\delta_1(q_0, 1, \#) = \{(q_0, 1\#)\}$$

$$\delta_1(q_0, 0, 0) = \{(q_0, 00), (q_1, \varepsilon)\}$$

$$\delta_1(q_0, 1, 0) = \{(q_0, 10)\}$$

$$\delta_1(q_0, 0, 1) = \{(q_0, 01)\}$$

$$\delta_1(q_0, 1, 1) = \{(q_0, 11), (q_1, \varepsilon)\}$$

$$\delta_1(q_1, 0, 0) = \{(q_1, \varepsilon)\}$$

$$\delta_1(q_1, 1, 1) = \{(q_1, \varepsilon)\}$$

$$\delta_1(q_1, \varepsilon, \#) = \{(q_2, \#)\}$$

$$\delta_1(q_0, \varepsilon, \#) = \{(q_2, \#)\}$$

$$\delta(q, a, Z) = \emptyset \quad \text{sonst}$$

$$\rightarrow L(M_1) = L$$

PDA – Beispiel (1), Berechnungen

... für 0110

$$(q_0, 0110, \#) \vdash (q_0, 110, 0\#) \vdash (q_0, 10, 10\#) \\ \vdash (q_1, 0, 0\#) \vdash (q_1, \varepsilon, \#) \vdash (q_2, \varepsilon, \#) \quad \text{✓}$$

$$(q_0, 0110, \#) \vdash (q_0, 110, 0\#) \vdash (q_0, 10, 10\#) \vdash (q_0, 0, 110) \vdash (q_0, \varepsilon, 0110) \quad \text{✗}$$

$$(q_0, 0110, \#) \vdash (q_2, 0110, \varepsilon) \quad \text{✗}$$

PDA – Beispiel (2)

$$L = \{ ww^R \mid w \in \{0,1\}^* \}$$

$M_2 = (\{q_0, q_1\}, \{0,1\}, \{0,1, \#\}, \delta_2, q_0, \#, \emptyset)$ mit

$$\delta_2(q_0, 0, \#) = \{(q_0, 0\#)\}$$

$$\delta_2(q_0, 1, \#) = \{(q_0, 1\#)\}$$

$$\delta_2(q_0, 0, 0) = \{(q_0, 00), (q_1, \varepsilon)\}$$

$$\delta_2(q_0, 1, 0) = \{(q_0, 10)\}$$

$$\delta_2(q_0, 0, 1) = \{(q_0, 01)\}$$

$$\delta_2(q_0, 1, 1) = \{(q_0, 11), (q_1, \varepsilon)\}$$

$$\delta_2(q_1, 0, 0) = \{(q_1, \varepsilon)\}$$

$$\delta_2(q_1, 1, 1) = \{(q_1, \varepsilon)\}$$

$$\delta_2(q_1, \varepsilon, \#) = \{(q_1, \varepsilon)\}$$

$$\delta_2(q_0, \varepsilon, \#) = \{(q_0, \varepsilon)\}$$

$$\delta(q, a, Z) = \emptyset \text{ sonst}$$

$$\rightarrow N(M_1) = L$$

Akzeptierende Zustand vs. leerer Keller

Satz 11.1. Zu jedem PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$ kann ein PDA $M' = (Q \cup \{q'_0, q_e\}, \Sigma, \Gamma \cup \{\$\}, \delta', q'_0, \$, F)$ mit $L(M) = N(M')$ konstruiert werden.

Beweis:

1. $\delta'(q'_0, \varepsilon, \$) = \{(q_0, \#\$)\}$ // Übergang in Anfangskonfiguration von M
2. Für alle $q \in Q, a \in \Sigma \cup \{\varepsilon\}, Z \in \Gamma$: $\delta(q, a, Z) \subseteq \delta'(q, a, Z)$ // Simulation von M
3. Für alle $f \in F, Z \in \Gamma \cup \{\$\}$: $(q_e, \varepsilon) \in \delta'(f, \varepsilon, Z)$ // möglicher Start der ...
4. Für alle $Z \in \Gamma \cup \{\$\}$: $(q_e, \varepsilon) \in \delta'(q_e, \varepsilon, Z)$ // ... Keller-Entleerung

➤ Markierung \$, falls M Keller leert, obwohl Eingabe nicht akzeptiert wird!

Akzeptierende Zustand vs. leerer Keller

$w \in L(M) \Leftrightarrow (q_0, w, \#) \vdash_M^* (f, \varepsilon, \gamma)$ für ein $f \in F$ und ein $\gamma \in \Gamma^*$

$\Leftrightarrow (q'_0, w, \$) \vdash_{M'} (q_0, w, \#\$)$ // mit Regel 1.

$\vdash_{M'}^* (f, \varepsilon, \gamma\$)$ // mit Regeln 2.

Dann gilt $(f, \varepsilon, \gamma\$) \vdash_{M'}^* (q_e, \varepsilon, \varepsilon)$ und somit $w \in N(M')$.

Da diese Berechnung nur möglich ist, falls $f \in F$,
gilt $w \in N(M')$ nur falls $w \in L(M)$.



Akzeptierende Zustand vs. leerer Keller

Satz 11.2. Zu jedem PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$ kann ein PDA M' mit $L(M') = N(M)$ konstruiert werden.

Beweisskizze: O.B.d.A. kann $F = \emptyset$ angenommen werden.

Setzen $M = (Q \cup \{q'_0, f\}, \Sigma, \Gamma \cup \{\$, \}, \delta', q'_0, \$, \{f\})$;

1. $\delta'(q'_0, \varepsilon, \$) = \{(q_0, \#\$)\}$ // Übergang in Anfangskonfiguration von M
2. Für alle $q \in Q, a \in \Sigma \cup \{\varepsilon\}, Z \in \Gamma$: $\delta(q, a, Z) = \delta'(q, a, Z)$ // Simulation
3. Für alle $q \in Q$: $(f, \varepsilon) \in \delta'(q, \varepsilon, \$)$ // Akzeptieren wenn M leeren Keller hat

$w \in L(M') \Leftrightarrow w \in N(M)$ analog zum Beweis von Satz 11.1.



Deterministische PDAs

Definition. Ein PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$ heißt **deterministisch (DPDA)**, wenn

1. $|\delta(q, a, Z)| \leq 1$ für alle $q \in Q, a \in \Sigma \cup \{\varepsilon\}, Z \in \Gamma$ und
2. wenn $\delta(q, \varepsilon, Z) \neq \emptyset$, dann $\delta(q, a, Z) = \emptyset$ für alle $a \in \Sigma$.

Beispiel: $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{A, \#\}, \delta, q_0, \#, \{q_0\})$ mit

$$\delta(q_0, a, \#) = (q_1, A\#)$$

$$\delta(q_1, a, A) = (q_1, AA)$$

$$\delta(q_1, b, A) = (q_2, \varepsilon)$$

$$\delta(q_2, b, A) = (q_2, \varepsilon)$$

$$\delta(q_2, \varepsilon, \#) = (q_0, \varepsilon)$$

$$\delta(q, a, Z) = \emptyset \text{ sonst}$$

$$L(M) = ?$$

PDAs vs. kfGs

Satz 11.3. Zu jeder kfG G kann ein PDA M konstruiert werden, sodass $N(M) = L(G)$ gilt.

Idee: Folgen Prinzip des Parsers für kfG

- Simulation einer Linksableitung im Keller
- Topsymbol ist am weitesten links stehendes Symbol der Satzform
 - falls Nichtterminal: mit nichtdeterministisch gewählter Regel ersetzen
 - falls Terminal: Vergleich mit nächstem Symbol der Eingabe
- akzeptieren mit leerem Keller
- benötigen keine Unterscheidung von Zuständen

PDA vs. kfGs

Satz 11.3. Zu jeder kfG G kann ein PDA M konstruiert werden, sodass $N(M) = L(G)$ gilt.

Beweis: Gegeben sei die kfG $G = (N, T, P, S)$. Konstruieren den PDA
 $M = (\{q\}, T, N \cup T, \delta, q, S, \emptyset),$

$\delta(q, a, a) = \{(q, \varepsilon)\}$ für alle $a \in T,$

$\delta(q, \varepsilon, A) = \{ (q, \alpha) \mid A \rightarrow \alpha \in P \}.$

Zeigen $S \xRightarrow{*}_L v\gamma$ in G gdw. $(q, v, S) \vdash^* (q, \varepsilon, \gamma)$ in M .

Dann folgt für alle $w \in T^*$: $S \xRightarrow{*} w$ gdw. $(q, w, S) \vdash^* (q, \varepsilon, \varepsilon),$
also $w \in L(G)$ gdw. $w \in N(M).$

Vollständige Induktion (1)

Für alle $i \geq 0$ gilt: Falls $(q, v, S) \vdash^i (q, \varepsilon, \gamma)$, dann $S \xRightarrow{*}_L v\gamma$:

- **I.A.:** $i = 0$. Gilt mit $v = \varepsilon$ und $\gamma = S$.
- **I.V.:** Falls $(q, v, S) \vdash^i (q, \varepsilon, \gamma)$, dann $S \xRightarrow{*}_L v\gamma$.
- **I.S.:** Angenommen $(q, v, S) \vdash^{i+1} (q, \varepsilon, \gamma)$.

1. Fall: Im letzten Schritt wurde eine Transition der Form $(q, \varepsilon) \in \delta(q, a, a)$ für ein $a \in T$ angewandt. Dann gibt es ein y , sodass $v = ya$ und $(q, ya, S) \vdash^i (q, a, a\gamma) \vdash (q, \varepsilon, \gamma)$. Somit gilt auch $(q, y, S) \vdash^i (q, \varepsilon, a\gamma)$. Nach I.V. gilt dann $S \xRightarrow{*}_L ya\gamma$, also $S \xRightarrow{*}_L v\gamma$.

Vollständige Induktion (1)

Für alle $i \geq 0$ gilt: Falls $(q, v, S) \vdash^i (q, \varepsilon, \gamma)$, dann $S \xRightarrow{*}_L v\gamma$:

- **I.A.:** $i = 0$. Gilt mit $v = \varepsilon$ und $\gamma = S$.
- **I.V.:** Falls $(q, v, S) \vdash^i (q, \varepsilon, \gamma)$, dann $S \xRightarrow{*}_L v\gamma$.
- **I.S.:** Angenommen $(q, v, S) \vdash^{i+1} (q, \varepsilon, \gamma)$.

2. Fall: Im letzten Schritt wurde eine Transition der Form $(q, \alpha) \in \delta(q, \varepsilon, A)$ für ein $A \rightarrow \alpha \in P$ angewandt. Dann gibt es ein γ' , sodass $\gamma = \alpha\gamma'$ und $(q, v, S) \vdash^i (q, \varepsilon, A\gamma') \vdash (q, \varepsilon, \alpha\gamma')$.

Nach I.V. gilt dann $S \xRightarrow{*}_L vA\gamma'$ und $vA\gamma' \Rightarrow_L v\alpha\gamma'$.

Somit gilt $S \xRightarrow{*}_L v\gamma$.



Vollständige Induktion (2)

Für alle $i \geq 0$ gilt: Falls $S \xRightarrow{i}_L v\gamma$, dann $(q, v, S) \vdash^* (q, \varepsilon, \gamma)$:

- **I.A.:** $i = 0$. Gilt mit $v = \varepsilon$ und $\gamma = S$.
- **I.V.:** Falls $S \xRightarrow{i}_L v\gamma$, dann $(q, v, S) \vdash^* (q, \varepsilon, \gamma)$.
- **I.S.:** Falls $S \xRightarrow{i+1}_L v\gamma$, dann gibt es ein $y \in T^*$ und eine Regel $A \rightarrow \alpha$ in P , so dass $S \xRightarrow{i}_L yA\gamma' \Rightarrow_L y\alpha\gamma'$.

Sei $\alpha = z\beta$, $z \in T^*$ und $\beta \in \{\varepsilon\} \cup N(N \cup T)^*$, sodass $yz = v$ und $\beta\gamma' = \gamma$.

Nach I.V.: $(q, y, S) \vdash^* (q, \varepsilon, A\gamma')$. Da $(q, \alpha) \in \delta(q, \varepsilon, A)$, gilt $(q, \varepsilon, A\gamma') \vdash (q, \varepsilon, \alpha\gamma')$ und $(q, \varepsilon, \alpha\gamma') = (q, \varepsilon, z\beta\gamma')$.

Mit $\delta(q, a, a) = \{(q, \varepsilon)\}$ für alle $a \in T$ gilt

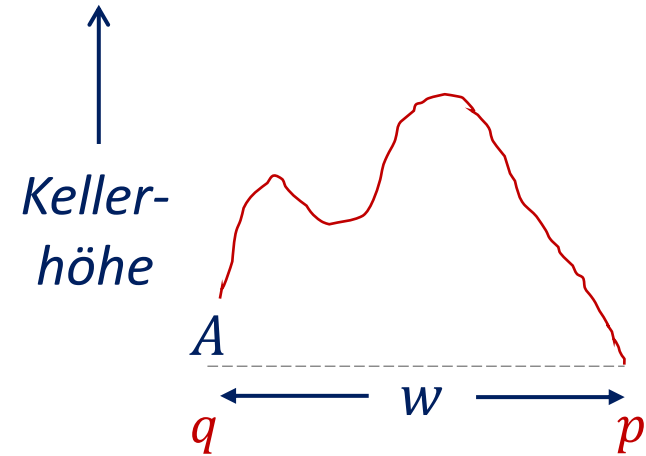
$(q, yz, S) \vdash^* (q, z, z\beta\gamma') \vdash^* (q, \varepsilon, \beta\gamma')$, also $(q, v, S) \vdash^* (q, \varepsilon, \gamma)$.

PDAs vs. kfGs

Satz 11.4. Zu jedem PDA M kann eine kfG G konstruiert werden, sodass $L(G) = N(M)$ gilt.

Idee:

- G hat Nichtterminale der Form $[q, A, p]$, $p, q \in Q, A \in \Gamma$, sodass der PDA M das Kellersymbol A vom Keller löscht, wenn er vom Zustand q in den Zustand p geht. Der Keller wird dabei nie unter dieses Vorkommen von A gelöscht.
- Aus dem Nichtterminal $[q, A, p]$ werden genau die terminalen Wörter w abgeleitet, für die $(q, w, A) \vdash^* (p, \varepsilon, \varepsilon)$ gilt.
- Die Menge der terminalen Wörter, die aus den Nichtterminalen der Form $[q_0, \#, p]$ abgeleitet werden (p beliebiger Zustand), ist die Sprache $N(M)$.

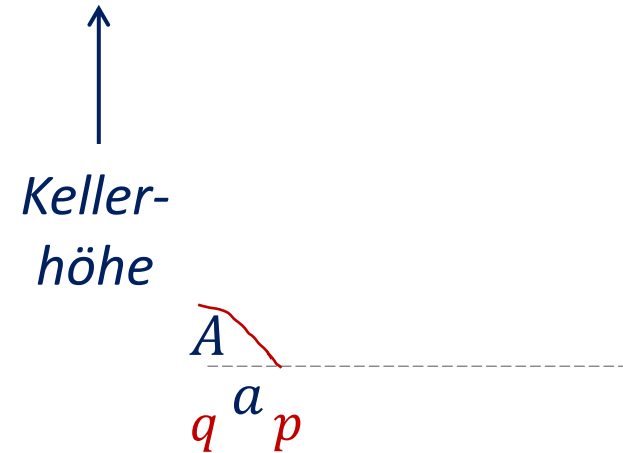


PDAs vs. kfGs

- *einfachster Fall:*

$$(p, \varepsilon) \in \delta(q, a, A) \quad (a \in \Sigma \cup \{\varepsilon\})$$

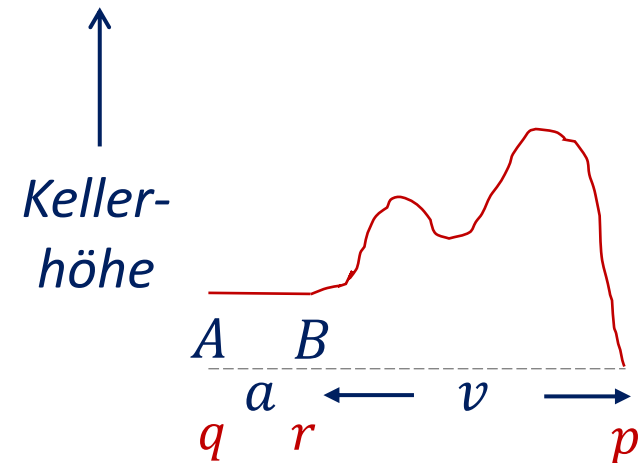
- Eingabe a lesen ist für M eine Möglichkeit, das Kellersymbol A zu löschen und dabei von q in p überzugehen
- Fügen Regel $[q, A, p] \rightarrow a$ hinzu.



- *Nächst einfacher Fall:*

$$(r, B) \in \delta(q, a, A) \quad (a \in \Sigma \cup \{\varepsilon\})$$

- Fügen Regel $[q, A, p] \rightarrow a[r, B, p]$ hinzu.

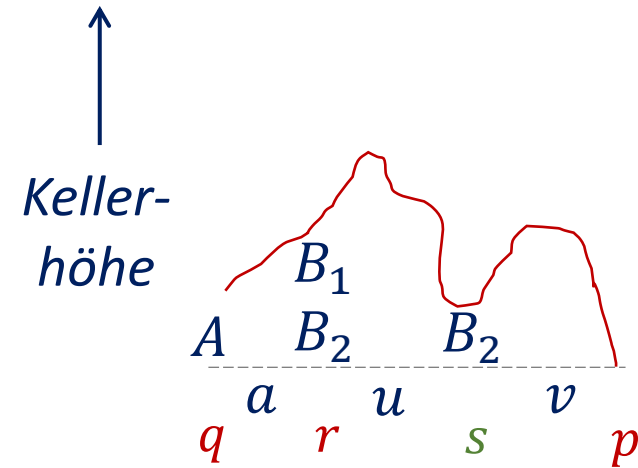


PDAs vs. kfGs

- *nächst einfacher Fall:*

$$(r, B_1 B_2) \in \delta(q, a, A) \quad (a \in \Sigma \cup \{\varepsilon\})$$

- A wird beim Lesen von a durch $B_1 B_2$ ersetzt. Dabei geht M in den Zustand r über.
- Nun muss B_1 durch Lesen eines Wortes u gelöscht werden, wobei M in einen Zustand s übergeht.
Dann muss B_2 durch Lesen eines Wortes v gelöscht werden, wobei M von s in p geht.
- Regeln $[q, A, p] \rightarrow a[r, B_1, s][s, B_2, p]$ für alle $s \in Q$



PDA vs. kfGs

Satz 11.4. Zu jedem PDA A kann eine kfG G konstruiert werden, sodass $L(G) = N(A)$ gilt.

Konstruktion: Gegeben ein PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, \#, F)$.
Konstruieren eine kfG $G = (V, \Sigma, P, S)$, wobei

- $V = \{S\} \cup \{ [q, A, p] \mid q, p \in Q, A \in \Gamma \}$ und
- P folgende Regeln enthält:
 - $S \rightarrow [q_0, \#, p]$ für alle $p \in Q$,
 - $[q, A, q_{m+1}] \rightarrow a[q_1, B_1, q_2][q_2, B_2, q_3] \dots [q_m, B_m, q_{m+1}]$ für alle $q, q_1, q_2, \dots, q_{m+1} \in Q$, $a \in \Sigma \cup \{\varepsilon\}$, $A, B_1, B_2, \dots, B_m \in \Gamma$, falls $(q_1, B_1 B_2 \dots B_m) \in \delta(q, a, A)$. (Falls $m = 0$: $[q, A, q_1] \rightarrow a$.)

Beweis
der Korrektheit
wieder durch zwei
Induktionsbeweise



PDA in kfG – Beispiel

$M = (\{q_0, q_1\}, \{a, b\}, \{A, \#\}, \delta, q_0, \#, \emptyset)$ mit

$$\delta(q_0, a, \#) = \{(q_0, A\#)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA)\}$$

$$\delta(q_0, b, A) = \{(q_1, \varepsilon)\}$$

$$\delta(q_1, b, A) = \{(q_1, \varepsilon)\}$$

$$\delta(q, a, Z) = \emptyset \text{ sonst}$$

$$\delta(q_1, \varepsilon, \#) = \{(q_1, \varepsilon)\}$$

$$L(M) = ?$$

$G = (N, T, P, S)$ mit

$$N = \{S, [q_0, A, q_0], [q_0, A, q_1], [q_1, A, q_0], [q_1, A, q_1], \\ [q_0, \#, q_0], [q_0, \#, q_1], [q_1, \#, q_0], [q_1, \#, q_1]\}$$

$$T = \{a, b\}$$

Geben nur die Regeln in P an, die von S aus erreichbar sind.

PDA in kfG – Beispiel

$$S \rightarrow [q_0, \#, q_0] \mid [q_0, \#, q_1]$$

$$\left. \begin{aligned} [q_0, \#, q_0] &\rightarrow a[q_0, A, q_0][q_0, \#, q_0] \\ [q_0, \#, q_0] &\rightarrow a[q_0, A, q_1][q_1, \#, q_0] \\ [q_0, \#, q_1] &\rightarrow a[q_0, A, q_0][q_0, \#, q_1] \\ [q_0, \#, q_1] &\rightarrow a[q_0, A, q_1][q_1, \#, q_1] \end{aligned} \right\}$$

wegen $\delta(q_0, a, \#) = \{(q_0, A\#)\}$

$$\left. \begin{aligned} [q_0, A, q_0] &\rightarrow a[q_0, A, q_0][q_0, A, q_0] \\ [q_0, A, q_0] &\rightarrow a[q_0, A, q_1][q_1, A, q_0] \\ [q_0, A, q_1] &\rightarrow a[q_0, A, q_0][q_0, A, q_1] \\ [q_0, A, q_1] &\rightarrow a[q_0, A, q_1][q_1, A, q_1] \end{aligned} \right\}$$

wegen $\delta(q_0, a, \#) = \{(q_0, AA)\}$

$$[q_0, A, q_1] \rightarrow b$$

$$[q_1, A, q_1] \rightarrow b$$

$$[q_1, \#, q_1] \rightarrow \varepsilon$$

wegen $\delta(q_0, b, A) = \{(q_1, \varepsilon)\}$

wegen $\delta(q_1, b, A) = \{(q_1, \varepsilon)\}$

wegen $\delta(q_1, \varepsilon, \#) = \{(q_1, \varepsilon)\}$

PDA in kfG – Beispiel

$$S \rightarrow [q_0, \#, q_0] \mid [q_0, \#, q_1]$$

$$[q_0, \#, q_0] \rightarrow a[q_0, A, q_0][q_0, \#, q_0]$$

$$[q_0, \#, q_0] \rightarrow a[q_0, A, q_1][q_1, \#, q_0]$$

$$[q_0, \#, q_1] \rightarrow a[q_0, A, q_0][q_0, \#, q_1]$$

$$[q_0, \#, q_1] \rightarrow a[q_0, A, q_1][q_1, \#, q_1]$$

$$[q_0, A, q_0] \rightarrow a[q_0, A, q_0][q_0, A, q_0]$$

$$[q_0, A, q_0] \rightarrow a[q_0, A, q_1][q_1, A, q_0]$$

$$[q_0, A, q_1] \rightarrow a[q_0, A, q_0][q_0, A, q_1]$$

$$[q_0, A, q_1] \rightarrow a[q_0, A, q_1][q_1, A, q_1]$$

$$[q_0, A, q_1] \rightarrow b$$

$$[q_1, A, q_1] \rightarrow b$$

$$[q_1, \#, q_1] \rightarrow \varepsilon$$

$$\begin{aligned} (q_0, aabb, \#) &\vdash (q_0, abb, A\#) \vdash (q_0, bb, AA\#) \\ &\vdash (q_1, b, A\#) \vdash (q_1, \varepsilon, \#) \vdash (q_1, \varepsilon, \varepsilon) \end{aligned}$$



$$\begin{aligned} S &\Rightarrow [q_0, \#, q_1] \\ &\Rightarrow a[q_0, A, q_1][q_1, \#, q_1] \\ &\Rightarrow aa[q_0, A, q_1][q_1, A, q_1][q_1, \#, q_1] \\ &\Rightarrow aab[q_1, A, q_1][q_1, \#, q_1] \\ &\Rightarrow aabb[q_1, \#, q_1] \\ &\Rightarrow aabb \end{aligned}$$

Zusammenfassung

Folgerung 11.5. Die folgenden drei Aussagen sind äquivalent:

1. L ist eine kontextfreie Sprache.
2. $L = L(M_1)$ für einen PDA M_1 .
3. $L = N(M_2)$ für einen PDA M_2 .