Äquivalenz von kontextfreien Sprachen und von Kellerautomaten

Prof. Dr. Jasmin Blanchette

Lehr- und Forschungseinheit für
Theoretische Informatik

Stand: 6. Juni 2023
Folien ursprünglich von PD Dr. David Sabel

Äquivalenz: PDAs und CFLs

- Wir zeigen, dass PDAs genau die Typ 2-Sprachen erkennen.
- Beweis in zwei Teilen:
  1. Konstruiere aus CFG in Greibach-Normalform einen PDA
  2. Konstruiere aus einem PDA eine CFG
     (sogenannte Tripelkonstruktion)

CFG $\to$ PDA (1)

Idee:

- CFG in Greibach-Normalform gegeben.
- PDA simuliert Linksableitung $S \Rightarrow^* w$.
- Da CFG in Greibach-Normalform, sieht eine Linksableitung nach i Schritten immer so aus:
$$S \Rightarrow^i a_1 \cdots a_i B_1 \cdots B_j$$
- Start mit Eingabe w und S auf dem Keller.
- Nach i Schritten, ist $a_1 \cdots a_i$ verarbeitet und $B_1 \cdots B_j$ auf dem Keller.

CFG $\to$ PDA (2)

Satz

Jede kontextfreie Sprache wird durch einen Kellerautomaten erkannt.

Beweis:

- Sei L eine CFL und $G = (V, \Sigma, P, S)$ mit $L(G) = L \setminus \{\varepsilon\}$ in Greibach-Normalform.
- Sei $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ ein PDA, sodass

$$\delta(z_0, a, A) := \{(z_0, B_1 \cdots B_n) \mid (A \to aB_1 \cdots B_n) \in P\}$$

  und falls $\varepsilon \in L$ setze zusätzlich $\delta(z_0, \varepsilon, S) := \{(z_0, \varepsilon)\}$.
  In allen anderen Fällen sei $\delta(z_0, \varepsilon, A) = \emptyset$.
- Wir zeigen $L(M) = L$.
- Zunächst: $\varepsilon \in L$ g.d.w. $(z_0, \varepsilon, S) \vdash (z_0, \varepsilon, \varepsilon)$ und damit $\varepsilon \in L(M)$.

CFG $\to$ PDA (3)

$M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit $\delta(z_0, a, A) := \{(z_0, B_1 \cdots B_n) \mid (A \to aB_1 \cdots B_n) \in P\}$ ...

Beweis (Fortsetzung):

- Für die weiteren Fälle zeigen wir für alle $i \in \mathbb{N}$ (mit Induktion über i)
  $S \Rightarrow^i_G a_1 \cdots a_i B_1 \cdots B_m$ mit einer Linksableitung g.d.w.
  $(z_0, a_1 \cdots a_i w, S) \vdash^i (z_0, w, B_1 \cdots B_m)$ für alle $w \in \Sigma^*$.
- Basis $i = 0$: Gilt, denn $S \Rightarrow^0_G S$ und $(z_0, w, S) \vdash^0 (z_0, w, S)$.
- Für $i > 0$ und „$\Rightarrow$“:
  - Sei $S \Rightarrow^i_G a_1 \cdots a_i B_1 \cdots B_m$ eine Linksableitung.
  - Da G in Greibach-Normalform, kann diese geschrieben werden als
    $S \Rightarrow^{i-1}_G a_1 \cdots a_{i-1} B_x B_{j+1} \cdots B_m \Rightarrow_G a_1 \cdots a_i B_1 \cdots B_m$,
    wobei $B_x \to a_i B_1 \cdots B_j \in P$ als letzte Produktion angewendet wurde.
  - Induktionsannahme liefert: $S \Rightarrow^{i-1}_G a_1 \cdots a_{i-1} B_x B_{j+1} \cdots B_m$ g.d.w.
    $(z_0, a_1 \cdots a_{i-1} w, S) \vdash^{i-1} (z_0, w, B_x B_{j+1} \cdots B_k)$.
  - Mit $w = a_i w'$, $(z_0, B_1 \cdots B_j) \in \delta(z_0, a_i, B_x)$, gilt $(z_0, a_1 \cdots a_i w', S) \vdash^i (z_0, w', B_1 \cdots B_k)$ für alle w'.

CFG → PDA (4)

$M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit $\delta(z_0, a, A) := \{(z_0, B_1 \cdots B_n) \mid (A \to aB_1 \cdots B_n) \in P\}$ ...

Beweis (Fortsetzung):

- Für $i > 0$ und „$\Leftarrow$“:
  - Sei $(z_0, a_1 \cdots a_i w, S) \vdash^i (z_0, w, B_1 \cdots B_k)$.
  - Dann muss der letzte Schritt a_i gelesen haben
  - D.h. die Folge lässt sich zerlegen in

    $$(z_0, a_1 \cdots a_i w, S) \vdash^{i-1} (z_0, a_i w, B_x B_{j+1} \cdots B_k) \vdash (z_0, w, B_1 \cdots B_k),$$

    wobei $(z_0, B_1 \cdots B_j) \in \delta(z_0, a_i, B_x)$.
  - Dann muss $B_x \to a_i B_1 \cdots B_j$ eine Produktion in P sein.
  - Induktionsannahme liefert: $S \Rightarrow_G^{i-1} a_1 \cdots a_{i-1} B_x B_{j+1} \cdots B_k$ und wir können obige Produktion anwenden und erhalten $S \Rightarrow_G^i a_1 \cdots a_i B_1 \cdots B_k$. □

Hilfssatz für PDA $\to$ CFG-Beweis

Lemma (PDAs mit Erzeugung von ≤ 2 Kellersymbolen)

Für jeden PDA $M = (Z, \Sigma, \Gamma, \delta, z_0, \#)$ gibt es einen PDA $M' = (Z, \Sigma, \Gamma', \delta', z_0, \#)$ mit $L(M') = L(M)$, sodass gilt: Wenn $(z', B_1 \cdots B_k) \in \delta'(z, a, A)$ (für $a \in \Sigma \cup \{\varepsilon\}$), dann ist $k \leq 2$.

Beweisskizze:
Transformiere M in M' wie folgt (mit $A \in \Gamma$ und $a \in \Sigma \cup \{\varepsilon\}$):

- $(z', B_1 \cdots B_k) \in \delta'(z, a, A)$, wenn $(z', B_1 \cdots B_k) \in \delta(z, a, A)$, $k \leq 2$.
- falls $(z', B_1 \cdots B_k) \in \delta(z, a, A)$ mit $k > 2$, dann
  - $(z, C_k B_k) \in \delta'(z, a, A)$, und
  - $\delta(z, \varepsilon, C_i) = \{(z, C_{i-1} B_{i-1})\}$ für alle i mit $4 \leq i \leq k$, und
  - $\delta(z, \varepsilon, C_3) = \{(z', B_1 B_2)\}$

  wobei $C_3, \ldots, C_k \in \Gamma'$ neue Kellersymbole sind
  (diese werden jeweils neu erzeugt pro ersetztem Eintrag). $\square$

PDA $\to$ CFG

Idee:

- Verwende PDA mit Erzeugung von ≤ 2 Kellersymbolen.
- Erzeuge Grammatik mit Tripelkonstruktion.
- Variablen der Grammatik:
  Tripel $\langle z', A, z\rangle$, die alle Wörter w erzeugt, die den PDA
  - von z' mit Kellerinhalt A und Wort w
  - zu z und leeren Keller führen.
- Produktionen (für $a \in \Sigma \cup \{\varepsilon\}$):

$\langle z', A, z\rangle \to a,$ wenn $z' \xrightarrow{(A, a) : \varepsilon} z$

$\langle z', A, z\rangle \to a\langle z'', B, z\rangle,$ wenn $z' \xrightarrow{(A, a) : B} z''$

$\langle z', A, z\rangle \to a\langle z'', B, z'''\rangle\langle z''', C, z\rangle,$ wenn $z' \xrightarrow{(A, a) : BC} z''$

PDA $\to$ CFG (2)

Satz

Kellerautomaten akzeptieren kontextfreie Sprachen.

Beweis: Sei $M = (Z, \Sigma, \Gamma, \delta, z_0, \#)$ ein PDA mit $k \leq 2$ für alle $(z', B_1 \cdots B_k) \in \delta(z, a, A)$ (und $a \in \Sigma \cup \{\varepsilon\}$).
Konstruiere $G = (V, \Sigma, P, S)$ mit S neues Symbol und

$$
\begin{array}{ll}
V = \{S\} \cup \{\langle z_i, A, z_j\rangle \mid z_i, z_j \in Z, A \in \Gamma\} & \\
P = \{S \to \langle z_0, \#, z\rangle \mid z \in Z\} & \\
\quad \cup \{\langle z', A, z\rangle \to a & \mid (z, \varepsilon) \in \delta(z', a, A), a \in \Sigma \cup \{\varepsilon\}, A \in \Gamma\} \\
\quad \cup \{\langle z', A, z\rangle \to a\langle z'', B, z\rangle & \mid (z'', B) \in \delta(z', a, A), z \in Z, a \in \Sigma \cup \{\varepsilon\}, A \in \Gamma\} \\
\quad \cup \{\langle z', A, z\rangle \to a\langle z'', B, z'''\rangle\langle z''', C, z\rangle & \mid (z'', BC) \in \delta(z', a, A), z, z''' \in Z, a \in \Sigma \cup \{\varepsilon\}, A \in \Gamma\}
\end{array}
$$

Wir beweisen $\langle z', A, z\rangle \Rightarrow_G^* w$ g.d.w. $(z', w, A) \vdash_M^* (z, \varepsilon, \varepsilon)$.

Da $S \to \langle z_0, \#, z\rangle \in P$ folgt: $w \in L(G) \iff w \in L(M)$, d.h. $L(G) = L(M)$.

PDA $\rightarrow$ CFG (3)

„$\Rightarrow$“:

- Sei $\langle z', A, z\rangle \Rightarrow_G^i w$ eine Linksableitung.
- Wir verwenden Induktion über i.
- Basis $i = 1$: Sei $\langle z', A, z\rangle \Rightarrow_G w$.
  - Verwendete Produktion muss $\langle z', A, z\rangle \rightarrow a$ sein.
  - Dann muss $(z, \varepsilon) \in \delta(z', a, A)$ gelten und damit gilt: $(z', a, A) \vdash (z, \varepsilon, \varepsilon)$.
- Induktionsschritt: Sei $\langle z', A, z\rangle \Rightarrow_G u \Rightarrow_G^{i-1} w$ mit $i - 1 > 0$.
  - Wenn $u = a \in \Sigma \cup \{\varepsilon\}$, dann kann $i - 1 > 0$ nicht gelten.
  - Wenn $u = a\langle z'', B, z\rangle$, dann $(z'', B) \in \delta(z', a, A)$ und $u = a\langle z'', B, z\rangle \Rightarrow^{i-1} aw' = w$. Dann gilt $\langle z'', B, z\rangle \Rightarrow^{i-1} w'$ und die Induktionsannahme liefert $(z'', w', B) \vdash_M^* (z, \varepsilon, \varepsilon)$. Mit $(z'', B) \in \delta(z', a, A)$ zeigt dies $(z', \underbrace{aw'}_{w}, A) \vdash_M (z'', w', B) \vdash_M^* (z, \varepsilon, \varepsilon)$.

PDA $\to$ CFG (4)

...

- Zur Erinnerung: Sei $\langle z', A, z\rangle \Rightarrow_G u \Rightarrow_G^{i-1} w$ mit $i-1>0$.
- Wenn $u = a\langle z'', B, z'''\rangle\langle z''', C, z\rangle$, dann ist $(z'', BC) \in \delta(z', a, A)$ und $u = a\langle z'', B, z'''\rangle\langle z''', C, z\rangle \Rightarrow^{i-1} aw' = w$.

  Dann gilt auch $\langle z'', B, z'''\rangle\langle z''', C, z\rangle \Rightarrow^{i-1} w'$ und es gibt Linksableitungen $\langle z'', B, z'''\rangle \Rightarrow^{j} w_0'$ und $\langle z''', C, z\rangle \Rightarrow^{k} w_1'$ mit $j+l \leq i-1$, $w' = w_0'w_1'$.

  Für beide können wir die Induktionsannahme anwenden und erhalten $(z'', w_0', B) \vdash_M^* (z''', \varepsilon, \varepsilon)$ und $(z''', w_1', C) \vdash_M^* (z, \varepsilon, \varepsilon)$.

  Abändern der 1. Konfigurationsfolge: C auf den Keller und w_1' anhängen $(z'', w', BC) = (z'', w_0'w_1', BC) \vdash_M^* (z''', w_1', C)$.

  Anhängen der 2. Konfigurationsfolge liefert: $(z'', w', BC) \vdash_M^* (z, \varepsilon, \varepsilon)$.

  Da $(z'', BC) \in \delta(z', a, A)$, gilt $(z', \underbrace{aw'}_{w}, A) \vdash_M (z'', w', BC) \vdash_M^* (z, \varepsilon, \varepsilon)$.

PDA $\to$ CFG (5)

„$\Leftarrow$“:

- Sei $(z', w, A) \vdash_M^i (z, \varepsilon, \varepsilon)$. Zeige $\langle z', A, z\rangle \Rightarrow_G^* w$ mit Induktion über i.
- Basis $i = 1$: Dann gilt $w = a \in \Sigma \cup \{\varepsilon\}$ und $(z, \varepsilon) \in \delta(z', a, A)$.
  Damit gibt es $\langle z', A, z\rangle \to a \in P$ und daher $\langle z', A, z\rangle \Rightarrow_G a$.
- Induktionsschritt: Sei $i > 1$ und daher $(z', aw', A) \vdash (z'', w', W) \vdash_M^{i-1} (z, \varepsilon, \varepsilon)$ für $i - 1 > 0$, $a \in \Sigma \cup \{\varepsilon\}$ und $W = \varepsilon$, $W = B$ oder $W = BC$.

  Wir betrachten alle drei Fälle für W einzeln:
  - $W = \varepsilon$: Dieser Fall ist nicht möglich, da $i - 1 > 0$ nicht gelten kann.
  - $W = B$. Dann ist $\langle z', A, z\rangle \to a\langle z'', B, z\rangle \in P$.
    Da $(z'', w', B) \vdash_M^{i-1} (z, \varepsilon, \varepsilon)$, liefert Induktionsannahme $\langle z'', B, z\rangle \Rightarrow_G^* w'$
    und daher: $\langle z', A, z\rangle \Rightarrow_G a\langle z'', B, z\rangle \Rightarrow_G^* aw' = w$.

PDA $\to$ CFG (6)

- ...
- $W = BC$. Dann ist $\langle z', A, z\rangle \to a\langle z'', B, z'''\rangle\langle z''', C, z\rangle \in P$.

  Schreibe $(z'', w', BC) \vdash_M^{i-1} (z, \varepsilon, \varepsilon)$ als
  $(z'', w_1' w_2', BC) \vdash_M^j (z''', w_2', C) \vdash_M^k (z, \varepsilon, \varepsilon)$ mit $j + k = i - 1$.

  Weglassen von C und w_2' im ersten Teil zeigt:
  $(z'', w_1', B) \vdash_M^j (z''', \varepsilon, \varepsilon)$.

  Da $j < i$ und $k < i$ liefert Induktionsannahme
  $\langle z'', B, z'''\rangle \Rightarrow_G^* w_1'$ und $\langle z''', C, z\rangle \Rightarrow_G^* w_2'$.

  Daher gilt

  $$\langle z', A, z\rangle \Rightarrow_G a\langle z'', B, z'''\rangle\langle z''', C, z\rangle \Rightarrow_G^* aw_1'\langle z''', C, z\rangle \Rightarrow_G^* \underbrace{a\,\underbrace{w_1' w_2'}_{w'}}_{w}. \qquad \square$$

Geschafft

Die gezeigten Sätze zusammengefasst ergeben:

Theorem

Kellerautomaten erkennen genau die kontextfreien Sprachen.

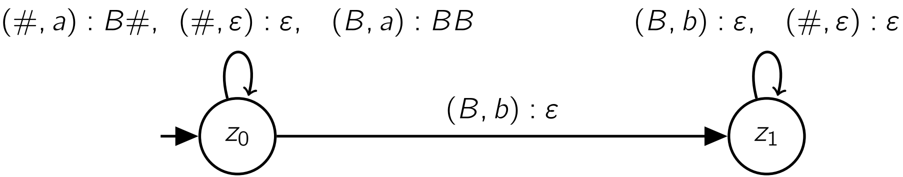
Bemerkung

Die bisherigen Beweise zeigen auch, dass man PDAs einschränken kann auf PDAs mit genau einem Zustand:

- Sei M ein PDA.
- Transformiere M in Grammatik G mit $L(G) = L(M)$.
- Transformiere G in G' in Greibach-Normalform (mit $L(G') = L(G) \setminus \{\varepsilon\}$).
- Transformiere Grammatik G' in PDA M' mit $L(M') = L(G)$.

  Die Konstruktion verwendet **nur einen Zustand**.

Beispiel (1)



Der vorherige Beweis konstruiert die Grammatik $G = (V, \Sigma, P, S)$ mit

$$
\begin{aligned}
V = \{ & S, \langle z_0, B, z_0\rangle, \langle z_0, B, z_1\rangle, \langle z_1, B, z_0\rangle, \langle z_1, B, z_1\rangle, \\
& \langle z_0, \#, z_0\rangle, \langle z_0, \#, z_1\rangle, \langle z_1, \#, z_0\rangle, \langle z_1, \#, z_1\rangle\} \\
P = \{ & S \to \langle z_0, \#, z_0\rangle, S \to \langle z_0, \#, z_1\rangle\} \\
& \cup \{\langle z_0, B, z_1\rangle \to b, \langle z_1, B, z_1\rangle \to b, \langle z_0, \#, z_0\rangle \to \varepsilon, \langle z_1, \#, z_1\rangle \to \varepsilon\} \\
& \cup \{\langle z_0, \#, z_0\rangle \to a\langle z_0, B, z_0\rangle\langle z_0, \#, z_0\rangle, \langle z_0, \#, z_0\rangle \to a\langle z_0, B, z_1\rangle\langle z_1, \#, z_0\rangle \\
& \quad \langle z_0, \#, z_1\rangle \to a\langle z_0, B, z_0\rangle\langle z_0, \#, z_1\rangle, \langle z_0, \#, z_1\rangle \to a\langle z_0, B, z_1\rangle\langle z_1, \#, z_1\rangle\} \\
& \cup \{\langle z_0, B, z_0\rangle \to a\langle z_0, B, z_0\rangle\langle z_0, B, z_0\rangle, \langle z_0, B, z_1\rangle \to a\langle z_0, B, z_0\rangle\langle z_0, B, z_1\rangle, \\
& \quad \langle z_0, B, z_0\rangle \to a\langle z_0, B, z_1\rangle\langle z_1, B, z_0\rangle, \langle z_0, B, z_1\rangle \to a\langle z_0, B, z_1\rangle\langle z_1, B, z_1\rangle\}
\end{aligned}
$$

Beispiel (2)

Umbenennen, Streichen von nicht erreichbaren Variablen und Entfernen von Einheitsproduktionen ergibt

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB,\ B \to b \mid aBC,\ C \to b\}, S)$$

(G ist bis auf ε-Produktion in Greibach-Normalform.)

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe $aaabbb$ ist

$$(z_0, aaabbb, S)$$

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe $aaabbb$ ist

$$\begin{aligned} & (z_0, aaabbb, S) \\ \vdash & (z_0, aabbb, B) \end{aligned}$$

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe *aaabbb* ist

$$\begin{aligned} & (z_0, aaabbb, S) \\ \vdash\ & (z_0, aabbb, B) \\ \vdash\ & (z_0, abbb, BC) \end{aligned}$$

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe $aaabbb$ ist

$$\begin{array}{l} (z_0, aaabbb, S) \\ \vdash (z_0, aabbb, B) \\ \vdash (z_0, abbb, BC) \\ \vdash (z_0, bbb, BCC) \end{array}$$

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe $aaabbb$ ist

$$\begin{array}{ll} & (z_0, aaabbb, S) \\ \vdash & (z_0, aabbb, B) \\ \vdash & (z_0, abbb, BC) \\ \vdash & (z_0, bbb, BCC) \\ \vdash & (z_0, bb, CC) \end{array}$$

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe *aaabbb* ist

$$\begin{array}{l} \ (z_0, aaabbb, S) \\ \vdash (z_0, aabbb, B) \\ \vdash (z_0, abbb, BC) \\ \vdash (z_0, bbb, BCC) \\ \vdash (z_0, bb, CC) \\ \vdash (z_0, b, C) \end{array}$$

Beispiel (3)

Der vorherige Beweis konstruiert für

$$G = (\{S, B, C\}, \{a, b\}, \{S \to \varepsilon \mid aB, B \to b \mid aBC, C \to b\}, S)$$

den PDA $M = (\{z_0\}, \Sigma, V, \delta, z_0, S)$ mit

$$\begin{array}{lll} \delta(z_0, a, S) = \{(z_0, B)\} & \delta(z_0, b, B) = \{(z_0, \varepsilon)\} & \delta(z_0, a, B) = \{(z_0, BC)\} \\ \delta(z_0, b, C) = \{(z_0, \varepsilon)\} & \delta(z_0, \varepsilon, S) = \{(z_0, \varepsilon)\} & \delta(z_0, d, A) = \emptyset \text{ sonst} \end{array}$$

Eine Konfigurationsfolge für die Eingabe *aaabbb* ist

$$\begin{array}{ll} & (z_0, aaabbb, S) \\ \vdash & (z_0, aabbb, B) \\ \vdash & (z_0, abbb, BC) \\ \vdash & (z_0, bbb, BCC) \\ \vdash & (z_0, bb, CC) \\ \vdash & (z_0, b, C) \\ \vdash & (z_0, \varepsilon, \varepsilon) \end{array}$$