Beh: L ist kontextfrei gdw L von einem PDA erkannt wird.

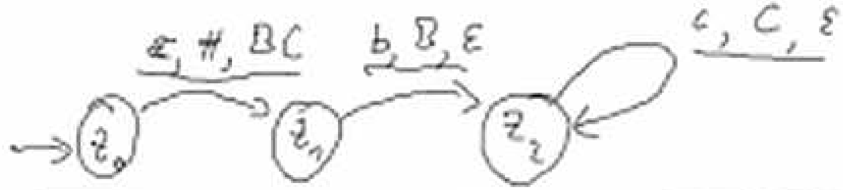
$\Rightarrow$ ✓

$\Leftarrow$ $L = N(M)$ mit $M = (Z, \Sigma, \Gamma, \delta, z_0, \#)$.

Konstruiere $G = (V, \Sigma, P, S)$

$V = \{S\} \cup Z \times \Gamma \times Z$

$$
\begin{aligned}
P = \Big(& \{S \to (z_0, \#, z) \mid z \in Z\} \;\cup \\
& \{(z, A, z') \to a \mid \delta(z, a, A) \ni (z', \varepsilon)\} \;\cup \\
& \{(z, A, z') \to a(z_1, B, z') \mid \delta(z, a, A) \ni (z_1, B),\ z' \in Z\} \;\cup \\
& \{(z, A, z') \to a(z_1, B, z_2)(z_2, C, z') \mid \delta(z, a, A) \ni (z_1, BC),\ z', z_2\} \Big)
\end{aligned}
$$



$L = \{ bc \}$

$$\boxed{z_0, bc, \# \vdash z_1, bc, BC \vdash z_2, c, C \vdash z_2, \varepsilon, \varepsilon}$$

$$S \to (z_0, \#, z_0) \mid (z_0, \#, z_1) \mid (z_0, \#, z_2)$$

$$(z_1, B, z_2) \to b$$

$$(z_2, C, z_2) \to c$$

$$\begin{aligned}(z_0, \#, z_0) \to\ & \varepsilon (z_1, B, z_0)(z_0, C, z_0) \mid \\ & \varepsilon (z_1, B, z_1)(z_1, C, z_0) \mid \\ & \varepsilon (z_1, B, z_2)(z_2, C, z_0)\end{aligned}$$

$$\begin{aligned}(z_0, \#, z_1) \to\ & \varepsilon (z_1, B, z_0), (z_0, C, z_1) \mid \\ & \varepsilon (z_1, B, z_1), (z_1, C, z_1) \mid \\ & \varepsilon (z_1, B, z_2), (z_2, C, z_1)\end{aligned}$$

$$\begin{aligned}(z_0, \#, z_2) \to\ & \varepsilon (z_1, B, z_2), (z_2, C, z_2) \\ & \vdots\end{aligned}$$

$$\begin{aligned}S & \Rightarrow (z_0, \#, z_2) \\ & \Rightarrow (z_1, B, z_2)(z_2, C, z_2) \\ & \Rightarrow b\ (z_2, C, z_2) \\ & \Rightarrow bc\end{aligned}$$

Beh.: $S \Rightarrow (z_0, \#, z') \Rightarrow^* x$ gdw. $z_0, x, \# \vdash^* z', \varepsilon, \varepsilon$ $\quad z' \in Z$

Wir zeigen $\forall\, x, A\ ,\ x \in \Sigma^*,\ A \in \Gamma$:

$$\underline{(z, A, z')} \Rightarrow^* \underline{x} \quad \text{gdw} \quad z, x, A \vdash^* \underline{z', \varepsilon, \varepsilon}$$

per Induktion über Anzahl der Ableitungsschritte bzw. Länge der Rechnung.

IA: 1 Schritt in der Abl. bzw. in der Rechnung

$$(z, A, z') \Rightarrow a \quad \text{gdw.} \quad ((z, A, z') \to a) \in P$$

$$\text{gdw.} \quad \delta(z, a, A) \ni (z', \varepsilon)$$

$$\text{gdw.} \quad z, a, A \vdash z', \varepsilon, \varepsilon$$

($\Leftarrow$): Für $n > 1$ Schritte sei $x = ay$ $\quad a \in (\Sigma \cup \{\varepsilon\})$:

$$\underbrace{z, ay, A \vdash z_1, y, \alpha}_{\text{der Schritt zusätzlich}} \vdash^* z', \varepsilon, \varepsilon$$

1) $\alpha = \varepsilon$, geht nicht, da z_1, y, ε keine Nachfolgekonf. hat.

2) $\alpha = B$

$z, ay, A \vdash \underbrace{z_1, y, B \vdash^* z', \varepsilon, \varepsilon}$

IV nutzen: $\underline{(z_1, B, z') \Rightarrow^* y}$

Außerdem für den 1. Schritt muss es ein Übergang ex.: $\delta(z, a, A) \ni (z_1, B)$, d.h. es gibt eine Regel $(z, A, z') \to a\,(z_1, B, z')$. D.h. $(z, A, z') \Rightarrow a\,(z_1, B, z') \Rightarrow ay = x$

3) $\alpha = BC$

$z, ay, A \vdash \underline{z_1, y, BC} \vdash^* z', \varepsilon, \varepsilon$

Wir zerlegen $y = y_1 y_2$, so dass $z_1, y, BC \vdash^* \underline{z_2, y_2, C}$.

Für y_1 muss gelten: $z_1, y_1, B \vdash^* z_2, \varepsilon, \varepsilon$. IV nutzen

$\underline{(z_1, B, z_2)} \Rightarrow^* y_1$. Nochmal IV: $\underline{(z_2, C, z')} \Rightarrow^* y_2$

Außerdem muss es eine Gramm-Regel geben

$(z, A, z') \to a\,\underline{(z_1, B, z_2)}\,\underline{(z_2, C, z')}$

D.h. $(z, A, z') \Rightarrow a\,(z_1, B, z_2)(z_2, C, z') \Rightarrow^* a y_1 y_2 = ay = x$.

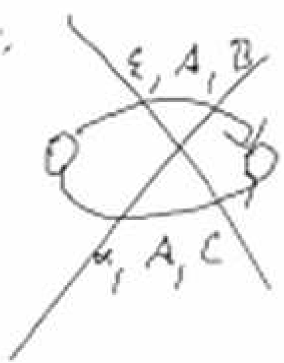
(⇐) entsprechend. $\square$

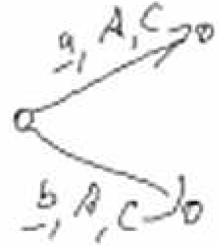
1.7.6 Deterministische kontextfreie Sprachen

<u>Def</u>. Ein <u>det. Kellerautomat</u> (DPDA) ist ein <u>PDAE</u>, bei dem für alle $z \in Z$, $a \in \Sigma$, $A \in \Gamma$ gilt

$$|\delta(z,a,A)| + |\delta(z,\varepsilon,A)| \le 1$$

<u>Bsp</u>:





<u>Bem</u>: Bei det. Kellerautomaten ist Akzeptanz durch leeren Keller nicht äquivalent mit Akzeptanz durch Endzustand und

Grund: Sprachen, bei denen echte Präfixe mit zur Sprache gehören.

$L = \{ (ab)^n \mid n \geq 1 \}$ $\quad L = \{ ab, abab, ababab, \dots \}$

Def Eine Sprache ist deterministisch kontextfrei, wenn sie von einem DPDA erkannt wird.

Bsp $L = \{ w \$ w^R \mid w \in \{a,b\}^* \}$ ist det. kf.

$L = \{ w w^R \mid w \in \{a,b\}^* \}$ ist nicht det. kf. !

Satz Die det. kf. Sprachen sind unter Komplement abgeschlossen.

Beweisidee: Im Prinzip wie bei DFAs (Endzustände und Nichtendzuständen vertauschen).

z_3, w, ε

z_3, w, ε und $z_3 \in E$

Satz Die det. kf. Sprachen sind nicht unter Schnitt und Vereinigung abgeschlossen.

Bew:

Schnitt: Bsp. für kf. Spr $L_1 = \{a^n b^k c^n d^l \mid k, n, l \geq 1\}$

$L_2 = \{a^k b^n c^l d^n \mid k, n, l \geq 1\}$

$L_1 \cap L_2$ ist nicht kf.

L_1 und L_2 sind mit DPDA erkennbar.

Vereinigung: (de Morgan und Widerspruchsbew.)

$L_1 \cap L_2 = \overline{\overline{L_1 \cap L_2}} = \overline{\overline{L_1} \cup \overline{L_2}}$. Angenommen die det. kf. sind unter Vereinigung abgeschlossen, dann müssen unter Schnitt abgeschlossen sein. ↯

$\square$

Bem: Die det. kf. Sprachen sind gerade die LR(k) Sprachen, die in linearer Zeit erkannt werden können.

1.8. Typ 1 und Typ 0 Sprachen

~~S → aSb~~

Automatenmodell?

→ Turingmaschine (Alan Turing 1912 – 1954)

→ zur Beschreibung allgemeiner Berechnungsprozesse