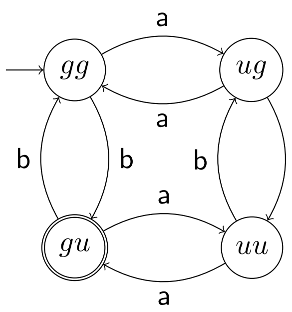
1) *Lösung.* Sei A der folgende DEA:



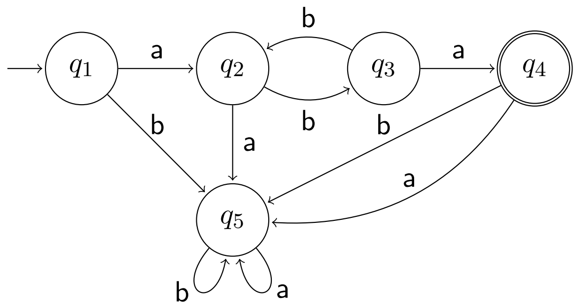
Somit gilt:

$$\begin{aligned}\hat{\delta}(\mathrm{gg}, \epsilon) &= \mathrm{gg} \notin F\\ \hat{\delta}(\mathrm{gg}, \mathsf{abab}) &= \cdots = \mathrm{gg} \notin F\\ \hat{\delta}(\mathrm{gg}, \mathsf{ababb}) &= \delta(\hat{\delta}(\mathrm{gg}, \mathsf{abab}), \mathsf{b}) = \delta(\mathrm{gg}, \mathsf{b}) = \mathrm{gu} \in F\ .\end{aligned}$$

Wir erhalten $\mathsf{ababb} \in \mathsf{L}(A)$ und $\epsilon, \mathsf{abab} \notin \mathsf{L}(A)$.

□

2) *Lösung.* Wir konstruieren den Automaten A, sodass $L = \mathsf{L}(A)$.



Einen DEA können wir im Allgemeinen in eine rechtslineare Grammatik $G = (V, \Sigma, R, S)$ transformieren, indem wir für jede Kante (p, a, q) des Automaten A eine Regel $P \to aQ$ in der Grammatik erzeugen, wobei $P, Q \in V$ und $a \in \Sigma$. Dabei ist folgendes zu beachten:

- Wenn p oder q ein Startzustand ist, so ist P oder Q das Startsymbol von G.
- Wenn q ein akzeptierender Zustand ist, fügen wir $Q \to \epsilon$ zu unseren Regeln hinzu.

- Das Eingabealphabet Σ von A ist das Alphabet Σ von G.
- Alle Kanten, die zu Zustände führen, von denen nie ein akzeptierender Zustand erreicht werden kann, können weggelassen werden.

Umgekehrt können wir auch aus einer rechtslinearen Grammatik G (unter gewissen Voraussetzungen) einen DEA $A = (Q, \Sigma, \delta, s, F)$ generieren, indem wir für jede Regel $P \to axQ$, wobei $P, Q \in V$, $a \in \Sigma$ und $x \in \Sigma^*$ die erweiterte Übergangsfunktion $\hat{\delta}(p, xa) = q$ definieren. Dabei ist folgendes zu beachten:

- Wenn P das Startsymbol ist, so ist p der Startzustand von A.
- Wenn Q leer ist, so ist q ein akzeptierender Zustand.
- Das Alphabet Σ von G ist das Eingabealphabet Σ von A.
- Für jedes unbekannte $\delta(q_1, a)$ generieren wir einen neuen Zustand $q_2 \in Q$ sodass $\delta(q_1, a) = q_2$ gilt.
- Alle nicht definierten kannten gehen zu einem neuen Zustand $r \in Q$.

Hinweis: Beachten Sie, dass im Allgemeinen die Konstruktion des Automaten aus einer (rechtslinearen) Grammatik einen sogenannten *nichtdeterministischen endlichen Automaten* (*NEA*)[1] generiert. Wir schränken uns aber auf solche Grammatiken ein, bei denen die resultierende Übergangsfunktion δ wohldefiniert ist. $\square$

3) *Lösung.* Im Basisfall ist $z = \epsilon$ und somit

$$\hat{\delta}(q, y\epsilon) = \hat{\delta}(q, y) = \hat{\delta}(\hat{\delta}(q, y), \epsilon)\,.$$

Im Induktionsschritt müssen wir

$$\hat{\delta}(q, yza) = \hat{\delta}(\hat{\delta}(q, y), za)$$

zeigen. Es gilt die Induktionshypothese

$$\hat{\delta}(q, yz) = \hat{\delta}(\hat{\delta}(q, y), z)\,.$$

Wir erhalten

$$\begin{aligned}
&\text{(Definition 4.15)} && \hat{\delta}(q, yza) = \delta(\hat{\delta}(q, yz), a)\\
&\text{(IH)} && \phantom{\hat{\delta}(q, yza)} = \delta(\hat{\delta}(\hat{\delta}(q, y), z), a)\\
&\text{(Definition 4.15)} && \phantom{\hat{\delta}(q, yza)} = \hat{\delta}(\hat{\delta}(q, y), za)\,.
\end{aligned}$$

$\square$

[1] Siehe https://de.wikipedia.org/w/index.php?title=Nichtdeterministischer_endlicher_Automat&oldid=213285292.