RWTH Aachen
Lehrgebiet Theoretische Informatik
Rossmanith–Dreier–Hark–Kuinke

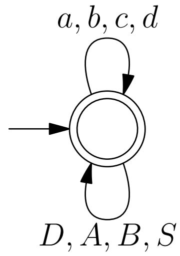
SS 2017
Blatt 7
19.6.2017

Lösungsvorschlag zur Vorlesung Formale Sprachen, Automaten und Prozesse

Aufgabe T21

a) Durch die beiden Regeln $S \Rightarrow ABC$ und $S \Rightarrow BCD$ sind schon alle Nichtterminale erreichbar.

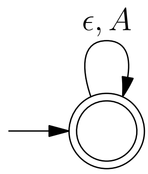
b) Die unproduktiven Symbole sind $N \setminus \mathit{pre}^*(T^*)$. Der Automat für $\mathit{pre}^*(T^*)$ lautet:



C ist das einzige Nichtterminal, welches nicht vom Automaten erkannt wird. Somit ist C unproduktiv.

c) Da S produktiv ist, kann von S ein $w \in T^*$ abgeleitet werden. Somit ist die von G erzeugte Sprache nicht leer.

d) Es gilt $\epsilon \in L(G)$ genau dann wenn $S \in \mathit{pre}^*(\{\epsilon\})$. Der Automat für $\mathit{pre}^*(\{\epsilon\})$ lautet:



Es gilt $\mathit{pre}^*(\{\epsilon\}) = \{\epsilon, A\}$. Somit enthält $L(G)$ nicht das leere Wort.

e) Die nullierbaren Symbole der Grammatik sind genau die Nichtterminale in $\mathit{pre}^*(\{\epsilon\})$. Diese Menge haben wir bereits in Teilaufgabe d) bestimmt, d.h. nur A läßt sich nach ϵ ableiten.

f) Regeln mit unerreichbaren oder unproduktiven Symbolen dürfen direkt gelöscht werden, da sie, in einer Ableitung vom Startsymbol aus, entweder nicht verwendet werden können (unerreichbar) oder bei Verwendung eine Ableitung eines Wortes aus Terminalsymbolen unmöglich machen (unproduktiv). Nullierbare Symbole können wie folgt behandelt werden:

1) Sei $A \to \epsilon \in G$ und $A \neq S$.
2) Lösche $A \to \epsilon$ aus G.

3) Für jede Regel $B \to \alpha A \beta$ füge die Regel $B \to \alpha\beta$ in G ein.
4) Falls keine Regel der Form $A \to \epsilon$ mehr existiert fertig, ansonsten gehe zu 1.

Somit hat die resultierende Grammatik keine Regeln der Form $A \to \epsilon$ für $A \neq S$. Die Regel $S \to \epsilon$ gehört genau dann zur Grammatik, wenn $\epsilon \in L(G)$ ist.

Für die Grammatik G ergibt sich also folgende Grammatik G':

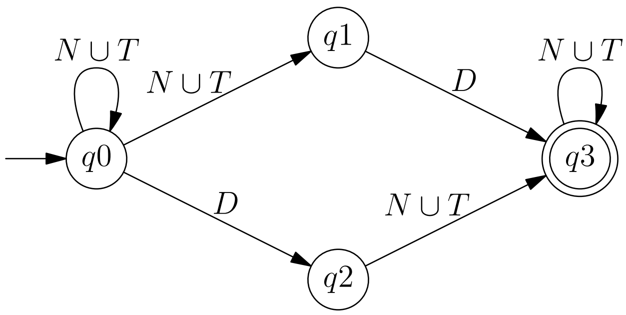
$$\begin{array}{lcl} S & \to & BD \\ A & \to & Sa \mid aB \\ B & \to & AS \mid S \mid aD \\ D & \to & bB \mid dD \mid b \mid c \mid d \end{array}$$

Aufgabe T22

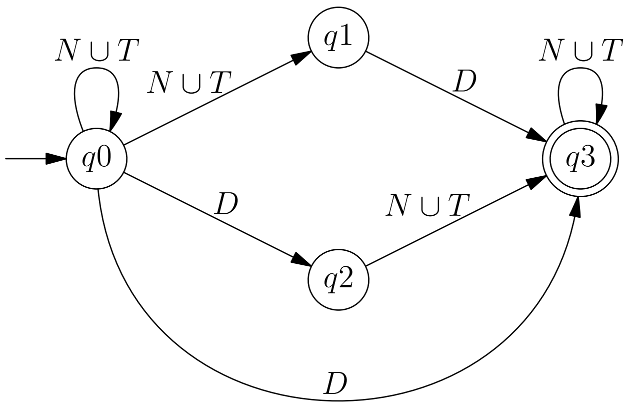
Um die Fragen zu beantworten, benötigen wir eine Grammatik G' mit $L(G') = L(G)$, die keine unproduktiven, unerreichbaren oder nullierbaren Symbole enthält. Die Grammatik G' aus Aufgabenteil T21 f) erfüllt diese Bedingung.

a) Aus der Vorlesung wissen wir: $L(G)$ ist genau dann endlich, falls es kein $A \in N$ gibt, so daß $A \in pre^*_{G'}(L)$, mit $L = (N \cup T)^+ A (N \cup T)^* \cup (N \cup T)^* A (N \cup T)^+$.

Wähle nun $A = D$. Wir konstruieren uns einen Automaten für L:



Nun können wir einen Saturierungsschritt für die Regel $D \to dD$ anwenden, (da $d \in N \cup T$) für die Transitionen $\delta(q_0, N \cup T) = q_1$ und $\delta(q_1, D) = q_3$, welches die Transition $\delta(q_0, D) = q_3$ liefert.

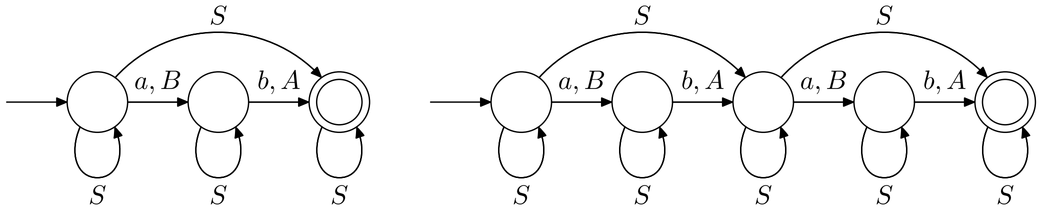


Demnach ist $D \in pre^*\left((N \cup T)^+ D (N \cup T)^* \cup (N \cup T)^* D (N \cup T)^+\right)$ und die sprache damit unendlich.

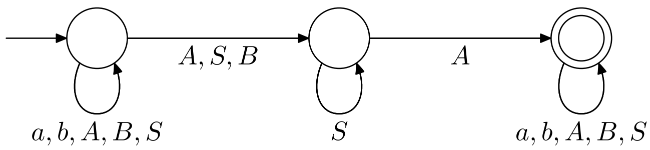
b) Nein, denn laut T21 d) gilt $\epsilon \notin L(G)$.

Aufgabe H16 (10 Punkte)

(a) Die Sättigung des NFA für $\{ab\}$ zeigt schon nach wenigen Schritten, daß S vom Start- in den Endzustand führt. Für $abab$ hingegen ergibt sich dies nicht. Es gilt also $ab \in L(G)$ und $abab \notin L(G)$.



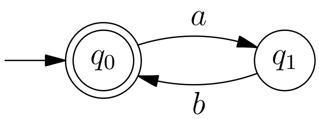
(b) Am gesättigten Automaten für $(a, b, A, B, S)^*AA(a, b, A, B, S)^*$ sieht man sofort, daß dies nicht möglich ist:



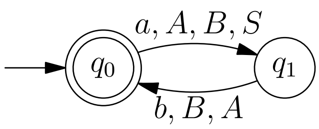
(c) Nein, denn ab läßt sich auf mehrere Weisen durch eine Linksableitung erzeugen, nämlich durch $S \Rightarrow aA \Rightarrow aSb \Rightarrow ab$ und $S \Rightarrow Bb \Rightarrow aSb \Rightarrow ab$.

Aufgabe H17 (4 Punkte)

Zunächst konstruieren wir einen NFA für die sprache $(ab)^*$:



Nun fügen wir den Sättigungsalgorithmus durch.



Jetzt können wir erkennen, daß SS nicht von q_0 nach q_0 führt, und wir somit die Frage mit nein beantworten können.