Prof. Dr. Andreas Podelski
Dr. Matthias Heizmann
Alexander Nutz

Abgabe: 19. Dezember 2018

9. Vorbereitungsblatt zur Veranstaltung
Informatik III

Dieses Vorbereitungsblatt soll Sie mit dem Beweis von Lemma vertraut machen. In diesem Beweis präsentieren wir ein Konstruktionsverfahren, dass zu einem NPDA $\mathcal{K}$ eine CFG $\mathcal{G}$ liefert sodass $L(\mathcal{G}) = L(\mathcal{K})$ gilt. Wir beschränken uns dabei auf NPDAs die in einem Berechnungsschritt höchtens ein zusätzliches Symbol auf den Stack legen. Formal: NPDAs mit $|\gamma| \leq 2$ für alle $(q', \gamma) \in \delta(q, x, Z), q, q' \in Q, x \in \Sigma \cup \{\varepsilon\}, Z \in \Gamma$.

Definiere $\mathcal{G} = (\Sigma, N, P, S)$ mit $N = (Q \times \Gamma \times Q) \cup \{S\}$. Wir wählen diese Tripel als Nichtterminalsymbole, da wir eine Grammatik konstruieren wollen, welche die folgende Eigenschaft hat:

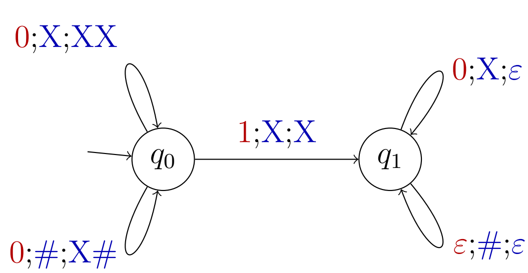
$$[q, Z, q'] \overset{n}{\vdash} w \text{ gdw } (q, w, Z) \rhd^n (q', \varepsilon, \varepsilon) \tag{$*$}$$

In Worten: Wir können vom Nichtterminalsymbol $[q, Z, q']$ das Wort w genau dann in n Schritten ableiten, wenn der NPDA die Konfiguration (q, w, Z) in n Schritten in eine akzeptierende Konfiguration mit Zustand q' überführen kann.
Als Menge der Regeln wählen wir $P = P_S \cup P_0 \cup P_1 \cup P_2$ mit:

$$\begin{aligned}
P_S &= \{S \to [q^{\mathsf{init}}, Z^{\mathsf{init}}, q'] \mid q' \in Q\}, \\
P_0 &= \{[q, Z, q'] \to x \mid (q', \varepsilon) \in \delta(q, x, Z), x \in \Sigma \cup \{\varepsilon\}\}, \\
P_1 &= \{[q, Z, q'] \to x[q'', Z', q'] \mid (q'', Z') \in \delta(q, x, Z), q' \in Q, x \in \Sigma \cup \{\varepsilon\}\}, \\
P_2 &= \{[q, Z, q'] \to x[q_1, Z_1, q_2][q_2, Z_2, q'] \mid (q_1, Z_1 Z_2) \in \delta(q, x, Z), q', q_2 \in Q, x \in \Sigma \cup \{\varepsilon\}\}
\end{aligned}$$

Dabei unterscheiden P_0, P_1 und P_2 die Fälle, dass der Stack kleiner wird (P_0), gleich hoch bleibt (P_1) oder größer wird (P_2).

Wir wollen diese Konstruktion nun an einem Beispiel betrachen und wählen dafür einen NPDA für die Sprache $L_{\text{centered3}} := \{0^n 1 0^n \mid n \in \mathbb{N}, n > 0\} \subseteq \{0, 1\}^*$, dessen Eingabealphabet $\Sigma = \{0, 1\}$ ist, dessen Kelleralphabet $\Gamma = \{\#, X\}$ ist und der das folgende Zustandsdiagramm hat.



Die folgende Sequenz von Berechnungsschritten zeigt beispielsweise dass das Wort 010 akzeptiert wird.

$$(q_0, 010, \#) \rhd (q_0, 10, X\#) \rhd (q_1, 0, X\#) \rhd (q_1, \varepsilon, \#) \rhd (q_1, \varepsilon, \varepsilon)$$

Entsprechend obiger Konstruktionsanletung erhalten wir die folgenden Regeln.

$$\begin{aligned}
P_S = \{ & S \to [q_0, \#, q_0] \\
& S \to [q_0, \#, q_1] \} \\
\\
P_0 = \{ & [q_1, X, q_1] \to 0 \\
& [q_1, \#, q_1] \to \varepsilon \} \\
\\
P_1 = \{ & [q_0, X, q_0] \to 1[q_1, X, q_0] \\
& [q_0, X, q_1] \to 1[q_1, X, q_1] \}
\end{aligned}
\qquad
\begin{aligned}
P_2 = \{ & [q_0, \#, q_0] \to 0[q_0, X, q_0][q_0, \#, q_0] \\
& [q_0, \#, q_0] \to 0[q_0, X, q_1][q_1, \#, q_0] \\
& [q_0, \#, q_1] \to 0[q_0, X, q_0][q_0, \#, q_1] \\
& [q_0, \#, q_1] \to 0[q_0, X, q_1][q_1, \#, q_1] \\
& [q_0, X, q_0] \to 0[q_0, X, q_0][q_0, X, q_0] \\
& [q_0, X, q_0] \to 0[q_0, X, q_1][q_1, X, q_0] \\
& [q_0, X, q_1] \to 0[q_0, X, q_0][q_0, X, q_1] \\
& [q_0, X, q_1] \to 0[q_0, X, q_1][q_1, X, q_1] \}
\end{aligned}$$

Aufgabe 1: Unnötige Regeln 2 Punkte

Entfernen Sie alle "unnötigen Regeln" aus der resultierenden Grammatik. Wir beschreiben diese Aufgabe präziser als das Entfernen alle Regeln sodass die folgenden zwei Eigenschaften gelten.

- Die Grammatik nach der Regelentfernung erzeugt die gleiche Sprache wie die Grammatik vor der Regelentfernung.
- Bei jeder weiteren Regelentfernung ändert sich die erzeugte Sprache.

Nutzen Sie zur Regelentfernung *keines* der in Vorlesung oder Übungen erwähnten Verfahren für beliebige CFGs sondern nutzen Sie Ihr Wissen über obigen NPDA und obiges Konstruktionsverfahren.

Sie dürfen gerne das Aufgabenblatt abgeben und darin die unnötigen Regeln durchstreichen.

Aufgabe 2: Ableitung 1 Punkt

Geben Sie in der resultirenden Grammatik eine Ableitung für das Wort 010 an.