Prof. Javier Esparza
Kevin Kappelmann, Simon Roßkopf, Lukas Stevens

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2023 – Übungsblatt 7

- Das Übungsblatt ist in zwei Teile gegliedert: den Vorbereitungsteil, den Sie vor der Übung selbstständig bearbeiten sollen, und den Übungs-/Nachbereitungsteil, der Aufgaben enthält, die in der Übung besprochen werden und von Ihnen anschließend zur Nachbereitung verwendet werden können.
- Für den Rest des Semesters gilt: $\mathbb{N} := \mathbb{N} := \{0, 1, 2, ...\}$ und $\mathbb{N}_+ := \mathbb{N} \setminus \{0\}$.

Notation von PDA-Regeln: Anstatt der in den Folien verwendeten Schreibweise $(q, YZ) \in \delta(p, a, X)$ für die Ersetzungsregeln eines PDA kann man alternativ $pX \xrightarrow{a} qYZ$ schreiben wobei $p, q \in Q$, $X \in \Gamma$, $YZ \in \Gamma^*$, $a \in \Sigma \cup \{\varepsilon\}$.

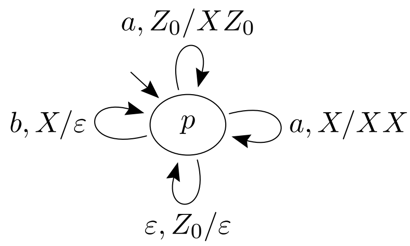
Beispiel: Den PDA mit δ:

$$\delta(p, a, Z_0) = \{(p, XZ_0)\} \qquad \delta(p, a, X) = \{(p, XX)\}$$
$$\delta(p, b, X) = \{(p, \varepsilon)\} \qquad \delta(p, \varepsilon, Z_0) = \{(p, \varepsilon)\}$$

schreibt man alternativ:

$$pZ_0 \xrightarrow{a} pXZ_0 \qquad pX \xrightarrow{a} pXX \qquad pX \xrightarrow{b} p \qquad pZ_0 \xrightarrow{\varepsilon} p$$

oder man stellt diesen als Graph mit Knotenmenge Q dar, wobei die Kante (p, q) dann mit “$a, X/YZ$” beschriftet ist:



Auf der folgenden Website können Sie PDAs konstruieren, simuliereren, testen,...
https://automatonsimulator.com/

Beachten Sie dabei: die PDAs auf der Website starten mit leerem Keller und akzeptieren mit Endzustand.

Vorbereitung (→ vor der Übung selbständig zu bearbeiten)

Individualaufgabe Ü7.1. (*Wichtige Begriffe*)

Überprüfen Sie, dass Sie die folgenden Begriffe oder Notationen korrekt definieren können.

- erzeugende, erreichbare, nützliche Nichtterminale

- CYK-Algorithmus
- Kellerautomat (PDA)
- Unterschied zwischen $L_\epsilon(A)$ und $L_F(A)$ für einen PDA A

Individualaufgabe Ü7.2. (*Automata Tutor: CYK & PDAs*)

Lösen Sie die Aufgaben Ü7.2 (a–f) auf Automata Tutor.

Achtung: Bei den *PDA construction* Aufgaben darf ihr konstruierter PDA nicht zu viele Zustände oder zu viele Stacksymbole haben (siehe Aufgabenstellung). Wenn Sie einen ε-Übergang angeben wollen, geben Sie statt ε bitte `E` ein (siehe Hinweisbox über Canvas). Die Simulation bei PDAs ist deaktiviert. Bitte wundern Sie sich nicht, dass bei einem Klick auf `Start Simulation` nichts passiert.

Tipp: Für den Aufgabentyp "CYK" können Sie sich zum Üben weitere Aufgaben von AT generieren lassen. Klicken Sie dafür auf `Home > My Autogenerated Problems` und wählen Sie den Aufgabentyp und gewünschten Schwierigkeitsgrad.

Lösungsskizze.

(a)

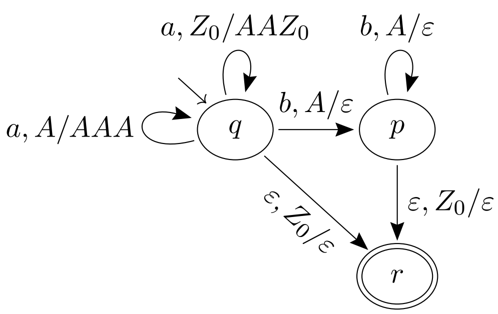
1,5 S				
1,4 S	2,5 S			
1,3	2,4 S	3,5		
1,2	2,3 L	3,4	4,5	
1,1 S	2,2 A	3,3 S	4,4 B	5,5 S
x	a	x	b	x

(b)

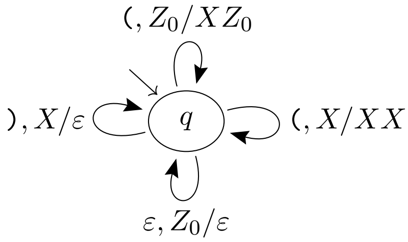
1,4 O			
1,3 O	2,4 Y		
1,2 O, U	2,3 Y	3,4 S	
1,1 O, K, U	2,2 O, K, U	3,3 S	4,4 S
d	d	v	v

(c) $\epsilon, bb, aababbab \in L$ und $abaab, bbb, aaabaaa \notin L$

(d) $abaababb, abba, ababbaaaba \in L$ und $baa, aab, bbbbb \notin L$



(e)

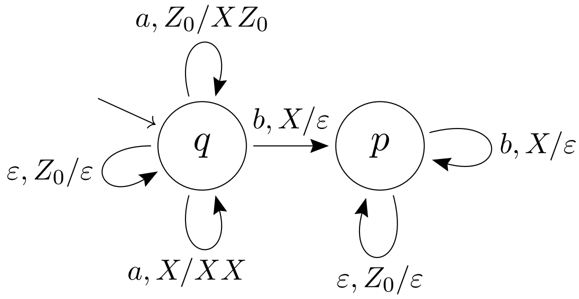


(f)

Individualaufgabe Ü7.3. (*(M)ein PDA*)

Zeichnen Sie einen PDA, der die Sprache $L = \{a^i b^i \mid i \geq 0\}$ erkennt.

Lösungsskizze. Idee: Für jedes a legen wir ein X auf den Stack und überprüfe dann, ob die die Anzahl von bs mit der Anzahl an X auf dem Stack übereinstimmt.
Ein PDA A der mit leerem Keller die Sprache L akzeptiert (also $L_\epsilon(A) = L$) ist:



Übung und Nachbereitung

Übungsaufgabe Ü7.4. (*Prächomsky-Normalform*)

Die CFG G bestehe aus folgenden Produktionen über dem Alphabet $\Sigma := \{a, b\}$:

$$\begin{array}{lcl} S & \to & ASA \mid aB \\ A & \to & B \mid S \mid CB \\ B & \to & b \mid \epsilon \\ C & \to & aC \\ D & \to & aSCb \mid a \end{array}$$

(a) Beschreiben Sie in eigenen Worten, wann ein Nichtterminal *nützlich* in einer Grammatik ist.

(b) Reduzieren Sie die Grammatik G auf die nützlichen Nichtterminale indem Sie zunächst alle nicht-erzeugende und anschließend alle nicht-erreichbare Nichtterminale eliminieren.

   Hinweis: Das Ergebnis ist die Grammatik aus Aufgabe Ü6.4.

Lösungsskizze.

(a) Ein Nichtterminal ist nützlich, wenn es eine Ableitung vom Startsymbol zu einem Wort (also einer Folge von Terminalen) gibt, in der das Nichtterminal verwendet wird. Nützliche Nichtterminale sind stets erzeugend und erreichbar. Erzeugend bedeutet, dass von dem

Nichtterminal ausgehend ein Wort produziert werden kann; erreichbar bedeutet, dass vom Startsymbol ausgehend das Nichtterminal produziert werden kann.

Der Begriff ist relevant, da ähnlich wie bei Automaten angenommen wird, dass alle Zustände erreichbar sind, man bei Grammatiken annehmen möchte, dass Sie nur nützliche Nichtterminale enthalten. Deswegen benötigt man eine klare Definition und eine Möglichkeit, eine Grammatik, die die Anforderung nicht erfüllt, in eine mit nur nützlichen Nichtterminalen zu überführen.

(b) *Erzeugende Nichtterminale*:

- Intuition: C kann kein Wort produzieren, da jede Ableitung von C beginnend stets mindestens ein Nichtterminal enthält.
- Formal bauen wir iterativ die Menge aller Nichtterminale, die eine Möglichkeit haben, ein Wort aus nur Terminalen zu erzeugen:
  (1) $P_0 := \{X \mid \exists (X, \gamma) \in P.\ \gamma \in \Sigma^*\}$
  (2) Wiederhole $P_{k+1} := P_k \cup \{X \in V \mid \exists (X, \gamma) \in P.\ \forall Y \in V.\ (|\gamma|_Y > 0 \to Y \in P_k)\}$ solange bis $P_{k+1} = P_k$.
- Führt auf:
$$P_0 = \{B, D\} \qquad P_1 = \{B, D, A, S\} = P_2$$
  Damit kann C samt $C \to aC$, $A \to CB$ und $D \to aSCb$ entfernt werden:
$$\begin{array}{lcl} S & \to & ASA \mid aB \\ A & \to & B \mid S \\ B & \to & b \mid \epsilon \\ D & \to & a \end{array}$$

Erreichbare Nichtterminale:

- Intuition: D kann von S aus nicht erreicht werden.
- Formal bauen wir die Menge aller Nichtterminale, die von S aus erreicht werden können:
  (1) $R_0 := \{S\}$
  (2) Wiederhole $R_{k+1} := R_k \cup \{Y \in V \mid \exists (X, \gamma) \in P.\ X \in R_k \wedge |\gamma|_Y > 0\}$ solange bis $R_{k+1} = R_k$.
- Führt auf:
$$R_0 = \{S\} \qquad R_1 = \{S, A, B\} = R_2$$
  Damit kann D und $D \to a$ entfernt werden.

Somit erhalten wir

$$S \to ASA \mid aB \qquad A \to B \mid S \qquad B \to b \mid \epsilon$$

Übungsaufgabe Ü7.5. (*CYK*)

(a) Beschreiben Sie in eigenen Worten, wie die Indizes in einer CYK-Tabelle zu verstehen sind.

(b) Beschreiben Sie in eigenen Worten, wie man den Inhalt eines Feldes in der CYK-Tabelle berechnet.

(c) Wir betrachten die Grammatik $G = (\{S, T, U, A, B, C\}, \{a, b, c\}, P, S)$ in CNF mit den folgenden Produktionen P:

$$
\begin{aligned}
S &\to TS \mid CT \mid a & \qquad A &\to a\\
T &\to AU \mid TT \mid c & \qquad B &\to b\\
U &\to SB \mid AB & \qquad C &\to c
\end{aligned}
$$

Bestimmen Sie mit dem CYK-Algorithmus, ob $ccaab \in L(G)$ und $aabcc \in L(G)$. Geben Sie dabei auch die berechneten Tabellen an.

Lösungsskizze.

(a) Das Feld $F_{x,y}$ beinhaltet alle Nichtterminale, die das Teilwort vom x-ten Terminal bis zum y-ten Terminal erzeugen können. (Wir fangen hier mit 1 an zu zählen.) Beispiel: Wenn das Wort $w = abaaab$ ist, dann beinhaltet $F_{2,4}$ die Terminale, die baa erzeugen.

(b) Für Felder $F_{x,x}$ (in der untersten Zeile) sucht man nach allen Nichtterminalen X, die eine Produktion $X \to a$ haben wobei a das x-te Zeichen des betrachteten Wortes ist.
Für Felder $F_{x,y}$ betrachtet man alle Kombinationen aus Feldern $F_{x,x} \times F_{x+1,y}$, $F_{x,x+1} \times F_{x+2,y}$ und so weiter.[1] Wenn (A, B) in einer der Mengen vorkommt und es eine Produktion $X \to AB$ gibt, dann fügt man X zum Feld $F_{x,y}$ hinzu.

(c) Nach dem CYK-Algorithmus ergeben sich folgende Berechnungstabellen:

1,5 S, T				
1,4	2,5 S, T			
1,3 S	2,4	3,5 T		
1,2 S, T	2,3 S	3,4	4,5 U	
1,1 C, T	2,2 C, T	3,3 S, A	4,4 S, A	5,5 B
c	c	a	a	b

1,5 S, T				
1,4 T	2,5			
1,3 T	2,4	3,5		
1,2	2,3 U	3,4	4,5 S, T	
1,1 S, A	2,2 S, A	3,3 B	4,4 C, T	5,5 C, T
a	a	b	c	c

Also ist $ccaab \in L(G)$ und $aabcc \in L(G)$.

[1] Tipp: Man legt seinen linken Zeigefinger auf das Feld $F_{x,x}$ und seinen rechten Zeigefinger auf $F_{x+1,y}$. Der linke Finger geht immer ein Feld höher, der rechte immer ein Feld nach unten rechts.

Übungsaufgabe Ü7.6. (*PDAs*)

Geben Sie für die folgenden Sprachen jeweils einen Kellerautomaten A_i in einer der oben aufgeführten Darstellungsarten an, sodass $L_i = L(A_i)$. Der Automat soll mit *leerem Stack* akzeptieren. Geben Sie dann zusätzlich für jeden Automaten jeweils ein nicht-leeres Wort w mit akzeptierendem Lauf an.

(a) $L_1 = \{a^n b^{3n} \mid n \geq 0\}$

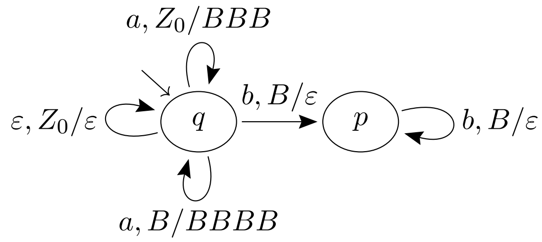
(b) $L_2 = \{a^n b^m \in \{a,b\}^* \mid n \leq m \leq 2n\}$

(c) $L_3 = \{w \in \{a,b\}^* \mid 2 \cdot |w|_a = 3 \cdot |w|_b\}$

Lösungsskizze.

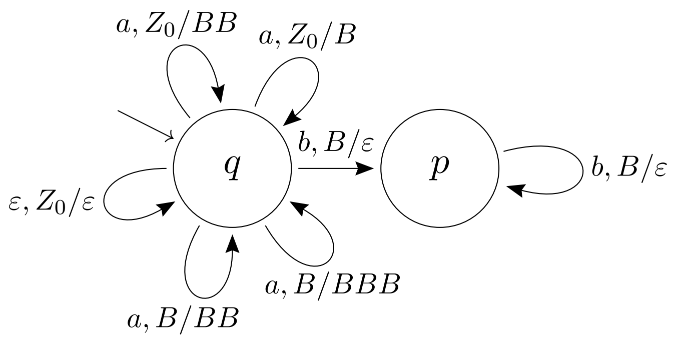
Wir geben die PDAs in Form einer Skizze des Automatens an. Für jeden der folgenden PDAs initialisieren wir den Stack mit dem Symbol Z_0.

(a)



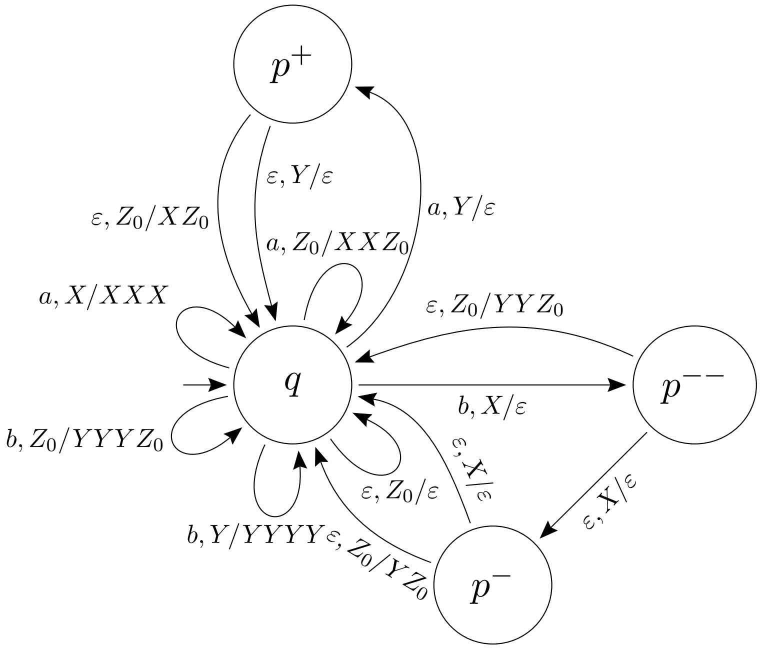
Akzeptierender Lauf: $(q, abbb, Z_0) \to (q, bbb, BBB) \to (p, bb, BB) \to (p, b, B) \to (p, \varepsilon, \varepsilon)$

(b) Idee: Für jedes a lege nichtdeterministisch entweder ein oder zwei b auf den Stack und überprüfe dann, ob die geratene Anzahl von bs mit der gegebenen übereinstimmt.



Akzeptierender Lauf: $(q, aabbb, Z_0) \to (q, abbb, BB) \to (q, bbb, BBB) \to (p, bb, BB) \to (p, b, B) \to (p, \varepsilon, \varepsilon)$

(c) Idee: Verwende Stack als (unären) Zähler und benutze explizites Bottom-Symbol (hier Z_0), um auf 0 zu testen. Für jedes a zähle um 2 (codiert als XX) hoch und für jedes b ziehe 3 (codiert als YYY) ab.



Akzeptierender Lauf: $(q, abbaa, Z_0) \to (q, bbaa, XXZ_0) \to (p^{--}, baa, XZ_0) \to (p^-, baa, Z_0) \to (q, baa, YZ_0) \to (q, aa, YYYYZ_0) \to (p^+, a, YYYZ_0) \to (q, a, YYZ_0) \to (p^+, \varepsilon, YZ_0) \to (q, \varepsilon, Z_0) \to (q, \varepsilon, \varepsilon)$