Prof. T. Nipkow
K. Kappelmann, J. Rädle, L. Stevens

KEINE ABGABE

Technische Universität München
Lehrstuhl für Logik und Verifikation

Einführung in die Theoretische Informatik

Sommersemester 2020 – Übungsblatt Lösungsskizze 5

AUFGABE 5.1. (*Wichtige Begriffe*) Stufe A

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

- nützlich, erzeugend, erreichbare Symbole
- Chomsky-Normalform
- CYK-Algorithmus
- Pumping-Lemma für kontextfreie Sprachen
- Präfix, Suffix, Infix

AUFGABE 5.2. (*Chomsky-Normalform*) Stufe C

Chomsky-Normalform Onlinetool

Sie können das Überführen von Grammatiken in Chomsky-Normalform auf der Website `http://grammar.epfl.ch/` üben. Beachten Sie dabei, dass die Website Feedback gibt, ob Ihre Lösung richtig oder falsch ist, aber nicht überprüft ob Sie das Verfahren aus der Vorlesung verwendet haben.

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

$$\begin{array}{lcl} S & \to & ASA \mid \mathsf{a}B \\ A & \to & B \mid S \mid CB \\ B & \to & \mathsf{b} \mid \varepsilon \\ C & \to & \mathsf{a}C \\ D & \to & \mathsf{a}SC\mathsf{b} \mid \mathsf{a} \end{array}$$

(a) Beschreiben Sie in eigenen Worten, wann ein Nichtterminal *nützlich* in einer Grammatik ist.
(b) Reduzieren Sie die Grammatik G auf die nützlichen Nichtterminale.
(c) Überführen Sie die reduzierte Grammatik dann in Chomsky-Normalform.
(d) Erklären Sie in eigenen Worten, wie Sie nach Überführen einer Grammatik in CNF überprüfen können, dass Sie keine Fehler gemacht haben.[1]

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 Symbole 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 Variablen*:
- Intuition: C kann kein Wort produzieren, da eine Ableitung, die ein C enthält, niemals "terminiert", also nie nur Terminale enthält.
- Formal bauen wir iterativ die Menge aller Nichtterminale, die eine Möglichkeit haben, ein Wort aus nur Terminalen zu erzeugen.
- $P_0 := \{X \mid \exists (X,\gamma) \in P.\ \gamma \in \Sigma^*\}$
- $P_{k+1} := P_k \cup \{X \in V \mid \exists (X,\gamma) \in P.\ \forall Y \in V.\ (|\gamma|_Y > 0 \to Y \in P_k)\}$ 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 $C \to \mathsf{a}C$, $A \to CB$ und $D \to \mathsf{a}SC\mathsf{b}$ entfernt werden.

Erreichbare Variablen:
- Intuition: D kann von S aus nicht erreicht werden.
- Formal bauen wir die Menge aller Nichtterminale, die von S aus erzeugt werden können.

[1] Hier geht es nicht um einen formalen Beweis, dass die beiden Sprachen gleich sind, sondern um eine Strategie, wie Sie bei Aufgaben wie im vorherigen Aufgabenteil Fehler vermeiden.

- $R_0 := \{\mathsf{S}\}$
- $R_{k+1} := R_k \cup \{Y \in V \mid \exists (X,\gamma) \in P.\ X \in R_k \wedge |\gamma|_Y > 0\}$ 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.

Ab jetzt steht G für die reduzierte (bereinigte) Grammatik:
$$S \to ASA \mid aB \qquad A \to B \mid S \qquad B \to b \mid \varepsilon$$

(c) *Überführung in CNF*:

- In jeder Regel (X,γ) mit $|\gamma| \geq 2$ Ersetzen jedes Vorkommen eines Terminals x durch X_x und Ergänzen der benötigten Produktionen $X_x \to x$:
$$\mathsf{G}' : S \to ASA \mid X_a B \qquad A \to B \mid S \qquad B \to b \mid \varepsilon \qquad X_a \to a$$
Alle rechten Seiten sind jetzt von der Form $VVV^* \cup \Sigma$.
- Überführen aller rechten Seiten, welche aus mindestens drei Variablen bestehen, in quadratische Monome über V. Die einfachste Variante ist dabei, aus XYZ einfach XX_{YZ} machen, wobei X_{YZ} einfach eine Hilfsvariable ist, die das Ergebnis von YZ mittels $X_{YZ} \to YZ$ zugewiesen bekommt. Damit:
$$\mathsf{G}'' : S \to AX_{SA} \mid X_a B \qquad A \to B \mid S \qquad B \to b \mid \varepsilon \qquad X_a \to a \qquad X_{SA} \to SA$$

Erkennen von ε-Regeln:

- $E_0 := \{X \in V \mid (X,\varepsilon) \in P\}$
- $E_{k+1} := E_k \cup \{X \in V \mid \exists (X,\gamma) \in P.\ \gamma \in E_k^*\}$ bis $E_{k+1} = E_k$.
$$E_0 = \{B\} \qquad E_1 = \{B, A\} = E_2$$
- Erzeugen zusätzlicher Produktionen, welche alle möglichen Kombinationen von ε-Produktionen beachten:
$$\begin{array}{lcl} S \to AX_{SA} & \rightsquigarrow & S \to AX_{SA} \mid X_{SA} \\ S \to X_a B & \rightsquigarrow & S \to X_a B \mid X_a \\ A \to B & \rightsquigarrow & A \to B \mid \varepsilon \\ A \to S & \rightsquigarrow & A \to S \\ B \to b & \rightsquigarrow & B \to b \\ B \to \varepsilon & \rightsquigarrow & B \to \varepsilon \\ X_a \to a & \rightsquigarrow & X_a \to a \\ X_{SA} \to SA & \rightsquigarrow & X_{SA} \to SA \mid S \end{array}$$
- Entfernen aller ε-Produktionen:
$$\mathsf{G}''' : S \to AX_{SA} \mid X_{SA} \mid X_a B \mid X_a \qquad A \to B \mid S \qquad B \to b \qquad X_a \to a \qquad X_{SA} \to SA \mid S$$

Zusammenziehen von Kettenproduktionen:

- $T_0 := \{(X,Y) \in P \cap V \times V\}$
- $T_{k+1} := T_k \cup \{(X,Y) \in V \times V \mid \exists Z \in V.\ (X,Z) \in T_k \wedge (Z,Y) \in T_k\}$ bis $T_{k+1} = T_k =: T_*$ (einfach transitiver Abschluss über durch Kettenproduktionen gegebener Kantenrelation auf V).
$$\begin{array}{lcl} T_0 & = & \{(S, X_{SA}), (S, X_a), (A, B), (A, S), (X_{SA}, S)\} \\ T_1 & = & T_0 \cup \{(S,S), (A, X_{SA}), (A, X_a), (X_{SA}, X_{SA}), (X_{SA}, X_a)\} = T_2 \end{array}$$
- Dann (1) Entfernen aller Kettenproduktion und (2) anschließend, falls $(X,Y) \in T_*$, füge (X,γ) zu P hinzu für jede Regel $(Y,\gamma) \in P$.
$$\begin{array}{lcl} \mathsf{G}'''' & : & S \to AX_{SA} \mid SA \mid X_a B \mid a \\ & & A \to b \mid AX_{SA} \mid SA \mid X_a B \mid a \\ & & B \to b \\ & & X_a \to a \\ & & X_{SA} \to SA \mid AX_{SA} \mid SA \mid X_a B \mid a \end{array}$$

(d)
- Prüfen, dass alle Produktionen die Form $\mathsf{A} \to \mathsf{BC}$ oder $\mathsf{A} \to \mathsf{a}$ haben.
- Insbesondere darf ε nur vom Startsymbol (oder gar nicht, wie hier) produziert werden.
- Für einige kleine/leichte Worte überprüfen, dass Sie von beiden Grammatiken (nicht) erzeugt werden können.

AUFGABE 5.3. (*CYK-Algorithmus*) Stufe C

Die folgende Aufgaben können Sie auch im AutomataTutor unter der Kategorie "CYK Algorithm" finden. Beachten Sie, dass sich dort auch eine Hausaufgabe mit limitierten Versuchen befindet.

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

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

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

Lösungsskizze
Nach dem CYK-Algorithmus ergeben sich folgende Berechnungstabellen:

15 S, T				
14	25 S, T			
13 S	24	35 T		
12 S, T	23 S	34	45 U	
11 C, T	22 C, T	33 S, A	44 S, A	55 B
c	c	a	a	b

15 S, T				
14 T	25			
13 T	24	35		
12	23 U	34	45 S, T	
11 S, A	22 S, A	33 B	44 C, T	55 C, T
a	a	b	c	c

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

AUFGABE 5.4. (*Pumping-Lemma für Kontextfreie Sprachen*) Stufe C

Wir betrachten Pfeil-Sprachen über dem Alphabet $\Sigma = \{\uparrow, \downarrow, \leftarrow, \rightarrow\}$. Wir interpretieren dabei ein Wort $w \in \Sigma^*$ als einen Pfad in einem 2D-Gitter.

(a) Geben Sie eine formale Definition für die folgenden beiden Sprachen an:
  (i) Pfade, die in den Ursprung zurückkehren.
  (ii) Pfade, die „in großer Kurve umkehren" — *beliebig weit* nach rechts fahren, dann *noch weiter* entweder nach oben oder unten gehen und letztlich wieder umkehren und *noch weiter* nach links fahren. Diese Pfade enden dann links vom Ursprung.

(b) Zeigen Sie mit Hilfe des Pumping-Lemma für kontextfreie Sprachen, dass diese Sprachen nicht kontextfrei sind

Lösungsskizze

(a) (i)

$$\mathsf{L}_a = \{w \in \Sigma^* \mid |w|_\uparrow = |w|_\downarrow \land |w|_\leftarrow = |w|_\rightarrow\}$$

(ii)

$$\mathsf{L}_b = \bigcup_{i<j<k} \mathsf{L}(\rightarrow^i (\uparrow^j \mid \downarrow^j) \leftarrow^k)$$

(b) (i)
- Wir nehmen an, dass L_a kontextfrei ist und führen diese Annahme zum Widerspruch.
- Sei $n \geq 1$ eine Pumping-Lemma-Zahl für die kontextfreie Sprache L_a.
- Sei zusätzlich $z = \rightarrow^n \uparrow^n \leftarrow^n \downarrow^n$, d.h., $z \in \mathsf{L}_a$ und $|z| \geq n$.
- Gemäß Pumping-Lemma gibt es dann also eine Zerlegung $z = uvwxy$ mit Wörtern $u, v, w, x, y \in \Sigma^*$ und den folgenden Eigenschaften:

$$(1)\ vx \neq \varepsilon \qquad (2)\ |vwx| \leq n \qquad (3)\ \forall i \in \mathbb{N}_0.\ uv^iwx^iy \in \mathsf{L}_a$$

- Aufgrund von (1) unterscheiden wir die folgenden Fälle:
  - $|vx|_\rightarrow > 0$: Wegen (2) gilt $|vx|_\leftarrow = 0$. Allerdings gilt auch:

$$\left|uv^2wx^2y\right|_\rightarrow = n + |vx|_\rightarrow > n = \left|uv^2wx^2y\right|_\leftarrow$$

    Dies ist ein Widerspruch zu (3).

- $|vx|_\uparrow > 0$: Wegen (2) gilt $|vx|_\downarrow = 0$. Allerdings gilt auch:

$$\left|uv^2wx^2y\right|_\uparrow = n + |vx|_\uparrow > n = \left|uv^2wx^2y\right|_\downarrow$$

Dies ist ein Widerspruch zu (3).
- $|vx|_\leftarrow > 0$: Wegen (2) gilt $|vx|_\rightarrow = 0$. Allerdings gilt auch:

$$\left|uv^2wx^2y\right|_\rightarrow = n < n + |vx|_\leftarrow = \left|uv^2wx^2y\right|_\leftarrow$$

Dies ist ein Widerspruch zu (3).
- $|vx|_\downarrow > 0$: Wegen (2) gilt $|vx|_\uparrow = 0$. Allerdings gilt auch:

$$\left|uv^2wx^2y\right|_\uparrow = n < n + |vx|_\downarrow = \left|uv^2wx^2y\right|_\downarrow$$

Dies ist ein Widerspruch zu (3).

- Da jeder Fall zu einem Widerspruch führt und die obigen Fälle alle möglichen Zerlegungen abdecken, kann die ursprüngliche Annahme nicht gelten. Also ist L_a nicht kontextfrei.

(ii)
- Wir nehmen an, dass L_b kontextfrei ist und führen diese Annahme zum Widerspruch.
- Sei $n \geq 1$ eine Pumping-Lemma-Zahl für die kontextfreie Sprache L_b.
- Dann ist $z = \rightarrow^n \uparrow^{n+1} \leftarrow^{n+2}$, d.h., $z \in \mathsf{L}_b$ und $|z| \geq n$.
- Gemäß Pumping-Lemma gibt es dann also eine Zerlegung $z = uvwxy$ mit Wörtern $u, v, w, x, y \in \Sigma^*$ und den folgenden Eigenschaften:

$$(1)\ vx \neq \varepsilon \qquad (2)\ |vwx| \leq n \qquad (3)\ \forall i \in \mathbb{N}_0.\ uv^iwx^iy \in \mathsf{L}_b$$

- Zuerst informell: Da $|vwx| \leq n$, kann vwx nur von der Form $\rightarrow^*\uparrow^*$ oder $\uparrow^*\leftarrow^*$ sein. Wegen $|vx| > 0$ muss mindestens ein Pfeil gepumpt werden. Gilt $|vx|_\rightarrow > 0$, dann können wir die Anzahl der $\rightarrow$ über die Anzahl der $\leftarrow$ pumpen, enthält vx keinen $\rightarrow$ aber mindestens ein $\uparrow$, so kann man die Anzahl der $\rightarrow$ auf höchstens n reduzieren, indem man vx entfernt. Andernfalls besteht vx nur aus $\leftarrow$, dann aus mindestens einem, so dass man durch Entfernen von vx die Anzahl der $\leftarrow$ auf $n+1$ oder weniger reduziert wird, man also höchstens so viele $\leftarrow$ wie $\uparrow$ hat.
- Formal: Aufgrund von (1) unterscheiden wir die folgenden Fälle:
  - $|vx|_\rightarrow > 0$: Wegen (2) gilt $|vx|_\leftarrow = 0$. Allerdings gilt auch:

$$\left|uv^3wx^3y\right|_\rightarrow = n + 2\,|vx|_\rightarrow \geq n + 2 = \left|uv^3wx^3y\right|_\leftarrow$$

  Dies ist ein Widerspruch zu (3).
  - $|vx|_\rightarrow = 0$ und $|vx|_\uparrow > 0$: Dann gilt:

$$\left|uv^0wx^0y\right|_\uparrow = n + 1 - |vx|_\uparrow \leq n = \left|uv^0wx^0y\right|_\rightarrow$$

  Daher ist $uv^0wx^0y \notin \mathsf{L}_b$, ein Widerspruch zu (3).
  - $|vx|_\rightarrow = 0$ und $|vx|_\uparrow = 0$: Dann muss $|vx|_\leftarrow > 0$ gelten, und es folgt:

$$\left|uv^0wx^0y\right|_\leftarrow = n + 2 - |vx|_\leftarrow < n + 1 = \left|uv^0wx^0y\right|_\uparrow$$

  Daher ist $uv^0wx^0y \notin \mathsf{L}_b$, ein Widerspruch zu (3).

Da jeder Fall zu einem Widerspruch führt und die obigen Fälle alle möglichen Zerlegungen abdecken, kann die ursprüngliche Annahme nicht gelten. Also ist L_b nicht kontextfrei.

AUFGABE 5.5.

Stufe D

Sei Σ ein Alphabet und $\mathsf{L} \subseteq \Sigma^*$. L ist *präfix-abgeschlossen*, wenn für alle $w \in \mathsf{L}$ und $v \in \Sigma^*$ mit $v \trianglelefteq w$ (also v ist ein Präfix von w) gilt: $v \in \mathsf{L}$.

Beweisen Sie, dass jede unendliche, präfix-abgeschlossene, kontextfreie Sprache eine unendliche, reguläre Teilmenge enthält.

Bemerkung: Zu dieser Aufgabe gibt es keine Videolösung. Eine textuelle Lösung finden Sie vorab auf Moodle/der Vorlesungswebsite.

Lösungsskizze
Sei L eine unendliche, präfix-abgeschlossene, kontextfreie Sprache. Sei $n \geq 1$ eine Pumping-Lemma-Zahl für die kontextfreie Sprache L. Da L unendlich ist, gibt es ein $z \in \mathsf{L}$ mit $|z| \geq n$. Gemäß Pumping-Lemma gibt es dann also eine Zerlegung $z = uvwxy$ mit Wörtern $u, v, w, x, y \in \Sigma^*$ und den folgenden Eigenschaften:

$$(1)\ vx \neq \varepsilon \qquad (2)\ |vwx| \leq n \qquad (3)\ \forall i \in \mathbb{N}_0.\ uv^iwx^iy \in \mathsf{L}$$

Wir unterscheiden nun zwei Fälle:
- Angenommen sei $v \neq \varepsilon$. Dann definieren wir nun die reguläre Sprache R_1:

$$R_1 = \mathsf{L}(uv^*) = \{uv^i \mid i \in \mathbb{N}_0\}$$

  R_1 ist unendlich und regulär. Wir haben dann $R_1 \subseteq \mathsf{L}$, wegen der Präfix-Abgeschlossenheit und (3).
- Angenommen sei $v = \varepsilon$. Aus (1) folgt dann $x \neq \varepsilon$. Wir definieren nun die reguläre Sprache R_2:

$$R_2 = \mathsf{L}(uwx^*) = \{uwx^i \mid i \in \mathbb{N}_0\}$$

  R_2 ist unendlich und regulär. Wir haben dann $R_2 \subseteq \mathsf{L}$, wegen der Präfix-Abgeschlossenheit und (3).

AUFGABE 5.6. (*Schnitt regelt*)
Beweisen Sie, dass kontextfreie Sprachen unter Schnitt mit regulären Sprachen abgeschlossen sind. Stufe D

Lösungsskizze
Sei L kontextfrei und L' regulär. Ohne Einschränkung gibt es $\mathsf{G} = (V, \Sigma, \mathsf{P}, \mathsf{S})$ CFG in CNF und $\mathsf{A} = (\mathsf{Q}, \Sigma, \delta, q_I, \mathsf{F})$ DFA mit $\mathsf{L} = \mathsf{L}(\mathsf{G})$ und $\mathsf{L}' = \mathsf{L}(\mathsf{A})$.
Idee: Wir erweitern G so, dass eine akzeptierende Berechnung von A geraten wird.
Definiere $\mathsf{G}' = (V', \Sigma, \mathsf{P}', \mathsf{S}')$ wie folgt:
- $V' = \mathsf{Q} \times V \times \mathsf{Q} \cup \{\mathsf{S}'\}$
- $\mathsf{P}' = \mathsf{P}'_{X \to YZ} \uplus \mathsf{P}'_{X \to \mathsf{a}} \uplus \mathsf{P}'_{\mathsf{S}'}$ mit

$$\begin{aligned} \mathsf{P}'_{X \to YZ} &:= \{(q, X, q') \to (q, Y, q'')(q'', Z, q') \mid X \to YZ \in P\} \\ \mathsf{P}'_{X \to \mathsf{a}} &:= \{(q, X, q') \to a \mid X \to a \in P \land \delta(q, a) = q'\} \\ \mathsf{P}'_{\mathsf{S}'} &:= \{\mathsf{S}' \to (q_I, \mathsf{S}, q_F) \mid q_F \in F\} \cup \{\mathsf{S}' \to \varepsilon \mid \varepsilon \in \mathsf{L}(\mathsf{G}) \cap \mathsf{L}(A)\} \end{aligned}$$

Behauptung: $\mathsf{L}(\mathsf{G}') = \mathsf{L}(\mathsf{G}) \cap \mathsf{L}(A)$
Sei $w = a_1 a_2 \dots a_l \in \mathsf{L}(\mathsf{G}')$. Dann gibt es eine Ableitung der Form

$$\mathsf{S}' \to (q_I, \mathsf{S}, q_F) \to^* (q_0, X_1, q_1)(q_1, X_2, q_2) \dots (q_{l-1}, X_l, q_l)$$

mit $(q_{i-1}, X_i, q_i) \to a_i$.
Nach Def. von $\mathsf{P}_{X \to YZ}$ muss $q_0 = q_I$ und $q_l = q_F$ gelten.
Nach Def. von $\mathsf{P}_{X \to \mathsf{a}}$ muss $q_i = \delta(q_{i-1}, a)$ gelten.
Damit existiert eine akzeptierende Berechnung von A zu w, also $w \in \mathsf{L}(A)$.
Nach Definition erhält man durch Vergessen der q, q', also durch Übergang von (q, X, q') zu X aus der obigen Ableitung auch eine Ableitung in G, womit auch $w \in \mathsf{L}(\mathsf{G})$ folgt.
Insgesamt $w \in \mathsf{L}(A) \cap \mathsf{L}(\mathsf{G})$.
Sei $w = a_1 a_2 \dots a_n \in \mathsf{L}(A) \cap \mathsf{L}(\mathsf{G})$. Dann existiert eine akzeptierende Berechnung $q_I = q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n \in F$ von A zu w und ein Ableitungsbaum zu w bzgl. G. Letzteren erweitert man induktiv zu einem Ableitungsbaum von G': Den inneren Knoten, an dem das Blatt zu a_i hängt, schreibt man von X auf (q_{i-1}, X, q_i) um. Bottom-up/induktiv schreibt man dann X nach (q_i, X, q_k) um, falls seine beiden Kinder (da CNF) gerade $(q_i, Y, q_j), (q_j, Z, q_k)$ sind. Da die Wurzel des konstruierten Ableitungsbaum (q_I, S, q_F) ist, muss man diese noch an eine neue Wurzel S' anhängen, um einen Ableitungsbaum bzgl. G' zu erhalten. Formal zeigt man die Aussage $X \to^*_{\mathsf{G}} w_i \dots w_j \Rightarrow (q_{i-1}, X, q_j) \to^*_{\mathsf{G}'} w_i \dots w_j$ per Induktion auf dem Ableitungsbaum bzgl. G.

Alternativer Beweis über Produkt von PDA und DFA:
Sei $\mathsf{P} = (Q_P, \Sigma, \Gamma, q_{0_P}, Z_0, \delta_P, F_P)$ ein PDA der L akzeptiert und $\mathsf{M} = (Q_M, \Sigma, \delta_M, q_{0_M}, F_M)$ ein DFA der L' akzeptiert.
Dann konstruieren wir einen PDA $P' = (Q, \Sigma, \Gamma, q_0, Z_0, \delta, F)$ der $\mathsf{L} \cap \mathsf{L}'$ akzeptiert wie folgt:
- $Q = Q_P \times Q_M$
- $q_0 = (q_{0_P}, q_{0_M})$
- $F = F_P \times F_M$
- $\delta((p, q), a, Z) = \{((p', q'), \beta) \mid (p', \beta) \in \delta_P(p, a, Z) \land q' = \delta_M(q, a)\}$

Beweisskizze: Es ist zu zeigen dass ein $w \in \Sigma^*$ genau dann von P' (mit Endzustand) akzeptiert wird wenn es in $\mathsf{L} \cap \mathsf{L}'$ liegt. Wir zeigen die folgende verallgemeinerte Aussage mit Induktion über w

$$(q_0, Z_0) \to^w_{P'} ((p, q), \sigma) \iff (q_{0_P}, Z_0) \to^w_P (p, \sigma) \land q_{0_M} \to^w_M q$$

Induktionsbasis: $w = \varepsilon$
Folgt aus der Definition von q_0 und δ.
Induktionsschritt: Sei $w' = wa$ für ein beliebiges $a \in \Sigma$. Mit der IH gilt $(q_0, Z_0) \rightarrow^{w}_{P'} ((p, q), \sigma) \iff (q_{0_P}, Z_0) \rightarrow^{w}_{P} (p, \sigma) \wedge q_{0_M} \rightarrow^{w}_{M} q$. Sei o.B.d.A $\sigma = Z\sigma'$ für ein $Z \in \Gamma$. Mit der Definition von δ gilt $((p, q), Z\sigma') \rightarrow^{a}_{P'} ((p', q'), \alpha\sigma') \iff (p, Z\sigma') \rightarrow^{w'}_{P} (p', \alpha\sigma') \wedge q_{0_M} \rightarrow^{w'}_{M} q'$. Mit der Induktionshypothese folgt die Behauptung.