Prof. Javier Esparza
Philipp Czerner, Martin Helfrich

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2021 – Übungsblatt 6

- Dies ist ein Entwurf. Die finale Version wird am Montag bereitgestellt, sich aber aller Wahrscheinlichkeit nicht besonders stark unterscheiden.
- 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.
- Das ist nicht das Hausaufgabenblatt! Die Hausaufgaben finden Sie auf einem separaten Blatt.

Vorbereitung (→ vor der Übung selbständig zu bearbeiten)

Individualaufgabe Ü6.1. (*Wichtige Begriffe*)

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

- nützlich, erzeugend, erreichbar (Symbole)
- Chomsky-Normalform
- Greibach-Normalform
- Pumping-Lemma für kontextfreie Sprachen
- Abschlusseigenschaften für kontextfreie Sprachen

Individualaufgabe Ü6.2. (*Automata Tutor: "Contextfree Languages"*)

Lösen Sie die Aufgaben Ü6.2 (a–b) auf Automata Tutor.[1] **Tipp:** Für die Aufgabentypen diese Woche 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) Zwischenschritte:

$$\begin{array}{l} S \to aSb \mid A \mid B \\ A \to aA \mid a \\ B \to bB \mid b \end{array} \quad \rightsquigarrow \quad \begin{array}{l} S \to FSG \mid A \mid B \\ A \to FA \mid a \\ B \to GB \mid b \\ F \to a \\ G \to b \end{array} \quad \rightsquigarrow \quad \begin{array}{l} S \to FR \mid A \mid B \\ R \to SG \\ A \to FA \mid a \\ B \to GB \mid b \\ F \to a \\ G \to b \end{array}$$

[1] Wenn Sie Automata Tutor noch nicht verwendet haben, folgen Sie erst den Schritten in Ü1.2, um sich richtig zu registrieren.

Lösung: $S \to FR \mid FA \mid a \mid GB \mid b$
$$
\begin{aligned}
R &\to SG\\
A &\to FA \mid a\\
B &\to GB \mid b\\
F &\to a\\
G &\to b
\end{aligned}
$$

(b) Zwischenschritte:

$$
\begin{array}{l}
S \to KDT \mid aT\\
K \to c \mid j \mid E\\
D \to bDS \mid abb\\
T \to \epsilon\\
E \to a \mid T
\end{array}
\rightsquigarrow
\begin{array}{l}
S \to KDT \mid AT\\
K \to c \mid j \mid ET \mid D\\
D \to BDS \mid ABB\\
T \to \epsilon\\
E \to a \mid T\\
A \to a\\
B \to b
\end{array}
\rightsquigarrow
\begin{array}{l}
S \to KX \mid AT\\
X \to DT\\
K \to c \mid j \mid ET \mid D\\
D \to BY \mid AZ\\
Y \to DS\\
Z \to BB\\
T \to \epsilon\\
E \to a \mid T\\
A \to a\\
B \to b
\end{array}
$$

$$
\rightsquigarrow
\begin{array}{l}
S \to KX \mid X \mid AT \mid A\\
X \to DT \mid D\\
K \to c \mid j \mid ET \mid D \mid E \mid T \mid \epsilon\\
D \to BY \mid AZ\\
Y \to DS\\
Z \to BB\\
T \to \epsilon\\
E \to a \mid T \mid \epsilon\\
A \to a\\
B \to b
\end{array}
\rightsquigarrow
\begin{array}{l}
S \to KX \mid X \mid A \mid A\\
X \to D \mid D\\
K \to c \mid j \mid E \mid D \mid E\\
D \to BY \mid AZ\\
Y \to DS\\
Z \to BB\\
E \to a\\
A \to a\\
B \to b
\end{array}
$$

Lösung: $S \to KX \mid BY \mid AZ \mid a$
$$
\begin{aligned}
X &\to BY \mid AZ\\
K &\to c \mid j \mid a \mid BY \mid AZ\\
D &\to BY \mid AZ\\
Y &\to DS\\
Z &\to BB\\
E &\to a\\
A &\to a\\
B &\to b
\end{aligned}
$$

Individualaufgabe Ü6.3. (*Pumping-Lemma für Kontextfreie Sprachen*)

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:

  (1) Pfade, die in den Ursprung zurückkehren.

  (2) 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. Für diese Aufgabe gibt es eine Video-Lösung.

(a) (1)

$$L_a = \{w \in \Sigma^* \mid |w|_\uparrow = |w|_\downarrow \wedge |w|_\leftarrow = |w|_\rightarrow\}$$

  (2)

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

(b) (1)
- 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., $z \in L_a$ 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:

$$(1)\ vx \neq \epsilon \qquad (2)\ |vwx| \leq n \qquad (3)\ \forall i \in N_0.\ uv^i wx^i y \in L_a$$

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

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

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

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

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

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

    Dies ist ein Widerspruch zu (3).

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

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

  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.

(2)
- 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., $z \in L_b$ 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:

$$(1)\ vx \neq \epsilon \qquad (2)\ |vwx| \leq n \qquad (3)\ \forall i \in N_0.\ uv^iwx^iy \in L_b$$

- 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:

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

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

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

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

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

    Daher ist $uv^0wx^0y \notin L_b$, 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.

Übung und Nachbereitung

Fokusaufgabe Ü6.4. (*Pumping-Lemma für Kontextfreie Sprachen*)

Entscheiden Sie ob die folgenden Sprachen kontextfrei sind. Wenn ja, geben Sie eine Grammatik an und zeigen Sie, dass Ihre Grammatik die Sprache akzeptiert. Wenn nein, beweisen Sie dies durch einen Widerspruchsbeweis unter Verwendung des Pumping Lemmas für kontextfreie Sprachen.

(a) $L_1 = \{a^n b^m c^l \mid n, m, l \in \mathbb{N} \wedge n > m \wedge n > l\}$

(b) $L_2 = \{ww \mid w \in \{a, b, c\}^* \wedge w = w^R\}$

Übungsaufgabe Ü6.5. (*Abschlusseigenschaften*)

Gegeben seien die kontextfreien Grammatiken $G_1 = (V_1, \Sigma_1, P_1, S_1)$ und $G_2 = (V_2, \Sigma_2, P_2, S_2)$. Konstruieren Sie aus diesen neue Grammatiken für die Sprachen:

(a) $L(G_1) \cup L(G_2)$

(b) $L(G_1)L(G_2)$

(c) $L^*(G_1)$

Lösungsskizze. Wir nehmen oBdA an, dass $V_1 \cap V_2 = \emptyset$, weil wir im Zweifel die Nichtterminale umbenennen, also z.B. mit 1 und 2 indexieren können. Sei $S \notin V_1 \cup V_2$ ein neues Nichtterminal.

(a) $G_\cup = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, P_1 \cup P_2 \cup \{S \to S_1 \mid S_2\}, S)$

(b) $G_{\text{konk}} = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, P_1 \cup P_2 \cup \{S \to S_1 S_2\}, S)$

(c) $G_{\text{stern}} = (V_1 \cup \{S\}, \Sigma_1, P_1 \cup \{S \to S_1 S \mid \epsilon\}, S)$

Übungsaufgabe Ü6.6. (*Chomsky-Normalform*)

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

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

(a) Überführen Sie die Grammatik in Chomsky-Normalform.

(b) Erklären Sie in eigenen Worten, wie Sie nach Überführen einer Grammatik in CNF überprüfen können, dass Sie keine Fehler gemacht haben.[2]

Lösungsskizze.

(a) *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 $X_x \to x$:

  $$G' \colon S \to ASA \mid X_a B \qquad A \to B \mid S \qquad B \to b \mid \varepsilon \qquad X_a \to a$$

  Alle rechten Seiten sind jetzt von der Form $V^* \cup \Sigma$.

[2] 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.

- 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:

$$G''\colon S \to AX_{SA} \mid X_aB \qquad A \to B \mid S \qquad B \to b \mid \varepsilon \qquad X_a \to a \qquad X_{SA} \to SA$$

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:

  $$\begin{array}{lcl} S \to AX_{SA} & \leadsto & S \to AX_{SA} \mid X_{SA} \\ S \to X_aB & \leadsto & S \to X_aB \mid X_a \\ A \to B & \leadsto & A \to B \mid \varepsilon \\ A \to S & \leadsto & A \to S \\ B \to b & \leadsto & B \to b \\ B \to \varepsilon & \leadsto & B \to \varepsilon \\ X_a \to a & \leadsto & X_a \to a \\ X_{SA} \to SA & \leadsto & X_{SA} \to SA \mid S \end{array}$$

- Entfernen aller ε-Produktionen:

  $$G'''\colon \begin{array}{l} S \to AX_{SA} \mid X_{SA} \mid X_aB \mid X_a \\ A \to B \mid S \\ B \to b \\ X_a \to a \\ X_{SA} \to SA \mid S \end{array}$$

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 $(Y, \gamma) \in P$.

  $$\begin{array}{lcl} G'''' & : & S \to AX_{SA} \mid SA \mid X_aB \mid a \\ & & A \to b \mid AX_{SA} \mid SA \mid X_aB \mid a \\ & & B \to b \\ & & X_a \to a \\ & & X_{SA} \to SA \mid AX_{SA} \mid SA \mid X_aB \mid a \end{array}$$

(b)
- Prüfen, dass alle Produktionen die Form $A \to BC$ oder $A \to a$ 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.