Theoretische Grundlagen der Informatik

Tutorium 7 — Wintersemester 2014/15

Moritz Klammler — 13. Januar 2015

Anmerkungen zum 4. Übungsblatt

- Um zu zeigen, dass $L \in \mathcal{NP}$, genügt es, einen polynomiellen *Verifizierer* anzugeben. Ein Suchalgorithmus ist nicht erforderlich.

Tagesthemen

- Zusammenfassung $\mathcal{P}$ / $\mathcal{NP}$
- Grammatiken
  - Reguläre Grammatiken
  - Kontextfreie Grammatiken
  - Kontextsensitive Grammatiken
  - Allgemeine Grammatiken

Grammatiken

Eine Grammatik G ist ein 4-Tupel $G = (T, V, S, P)$ bestehend aus

1. einem **Terminalalphabet** T,
2. einem dazu disjunkten **Nichtterminalalphabet** V,
3. einem **Startsymbol** $S \in V$ und
4. einer endlichen Menge an **Produktionen**[1] $P \subset (V \cup T)^+ \times (V \cup T)^*$.

[1] Diese Definition ist „mangelhaft“.

Reguläre Grammatiken

Sei $G = (T, V, S, P)$ eine Grammatik. G heißt **regulär** genau dann wenn für alle $(\alpha, \beta) \in P$ gilt, dass

- $\alpha \in V$ und
- $\beta \in \{\epsilon\} \cup TV$.

Reguläre Grammatiken heißen auch **Chomsky Typ-3 Grammatiken**.

Reguläre Grammatiken

Reguläre Grammatiken erzeugen genau die …

- … regulären …
- … von endlichen Automaten erkannten …
- … von regulären Ausdrücken beschriebenen …

… Sprachen.

Kontextfreie Grammatiken

Sei $G = (T, V, S, P)$ eine Grammatik. G heißt **kontextfrei** genau dann wenn für alle $(\alpha, \beta) \in P$ gilt, dass

- $\alpha \in V$ und
- $\beta \in (V \cup T)^*$.

Kontextfreie Grammatiken heißen auch **Chomsky Typ-2 Grammatiken**.

Kontextfreie Grammatiken

Kontextfreie Grammatiken erzeugen genau die …

- … kontextfreien …
- … von nichtdeterministischen Kellerautomaten erkannten …

… Sprachen.

Chomsky-Normalform

BONUS

Sei $G = (T, V, S, P)$ eine kontextfreie Grammatik. G ist in **Chomsky-Normalform (CNF)** genau dann wenn für alle $(\alpha, \beta) \in P$ gilt, dass

- $\alpha \in V$ und
  - $\beta \in V^2 \cup T$ oder
  - $\alpha = S$ und $\beta = \epsilon$.

Für jede kontextfreie Grammatik $G_1 = (T, V_1, S_1, P_1)$ existiert eine kontextfreie Grammatik $G_2 = (T, V_2, S_2, P_2)$ in Chomsky-Normalform, sodass $L(G_1) = L(G_2)$.

Kontextfreie Grammatik in CNF überführen

BONUS

1. Falls $(S, \epsilon) \in P$: Neues Startsymbol S' einführen und neue Produktion $S' \to S \mid \epsilon$ hinzunehmen.
2. Terminale $\mathtt{a}$ in nicht-terminierenden rechten Seiten durch neue Nichtterminale $X_{\mathtt{a}}$ ersetzen und neue Produktionen $X_{\mathtt{a}} \to \mathtt{a}$ hinzunehmen.
3. Wiederholen bis nicht mehr anwendbar: $A \to BCD$ ersetzen durch $A \to Y_{BC}D$ und neue Produktion $Y_{BC} \to BC$ hinzunehmen.
4. ϵ-Produktionen $A \to \epsilon$ (für $A \neq S$) streichen und für $L \to AR$ neue Produktion $L \to R$ hinzunehmen.
5. Kettenregeln $A \to B$ streichen und für $B \to w$ neue Produktion $A \to w$ hinzunehmen.

BONUS

Cocke-Younger-Kasami-Algorithmus

```
;; Eingabe: Grammatik G = (T,V,S,P) in CNF und Wort w ∈ T* mit |w| = n
;; Ausgabe: Aussage ob w ∈ L(G)
FOR i ← 1,...,n
    table[i][1] ← ∅
    FOREACH (l,r) ∈ P ;; terminierende Produktion
        IF r = w[i]
            table[i][1] ← table[i][1] ∪ {l}
FOR j ← 2,...,n
    FOR i ← 1,...,n − j
        table[i][j] ← ∅
        FOR k ← 1,...,j
            FOREACH (l,(r1,r2)) ∈ P ;; nicht−terminierende Produktionen
                IF r1 ∈ table[i][k] ∧ r2 ∈ table[i + k][j − k]
                    table[i][j] ← table[i][j] ∪ {l}
RETURN S ∈ table[1][n]
```

Pumping-Lemma für kontextfreie Sprachen

BONUS

Sei Σ ein Alphabet und $L \subset \Sigma^*$ eine kontextfreie Sprache.

Dann existiert ein $p \in \mathbb{N}$ sodass für alle $w \in L$ mit $|w| \geq p$ gilt: es existieren $u, v, x, y, z \in \Sigma^*$, sodass

1. $w = xuyvz$,
2. $|uv| \geq 1$,
3. $|uyv| \leq p$ und
4. $\forall i \in \mathbb{N}_0 : xu^i yv^i z \in L$.

Kontsensitive Grammatiken

Sei $G = (T, V, S, P)$ eine Grammatik.

G heißt **kontextsensitiv** genau dann wenn für alle $(\alpha, \beta) \in P$ gilt, dass

- $\alpha \in V^+$ und
  - $\beta \in (V \setminus \{S\} \cup T)^*$ und $|\alpha| \leq |\beta|$ oder
  - $\alpha = S$ und $\beta = \epsilon$.

Kontextsensitive Grammatiken heißen auch **Chomsky Typ-1 Grammatiken**.

Kontextsensitive Grammatiken

Kontextsensitive Grammatiken erzeugen genau die …

- … kontextsensitiven …
- … von nichtdeterministischen[2] linear beschränkten Turingmaschinen erkannten …

… Sprachen.

[2] Es ist unklar, ob das auch für deterministische gilt.

Kontextsensitive Normalform

BONUS

Sei $G = (T, V, S, P)$ eine kontextsensitive Grammatik. G ist in **Normalform** genau dann wenn für alle $(\alpha, \beta) \in P$ gilt, dass

- $\alpha = S$ und $\beta = \epsilon$ oder
- $\alpha \in V$ und $\beta \in (V \setminus \{S\}) \cup (V \setminus \{S\})^2 \cup T$ oder
- $\alpha \in V^2$ und $\beta \in (V \setminus \{S\})^2$.

Für jede kontextsensitive Grammatik $G_1 = (T, V_1, S_1, P_1)$ existiert eine kontextsensitive Grammatik $G_2 = (T, V_2, S_2, P_2)$ in Normalform, sodass $L(G_1) = L(G_2)$.

Allgemeine Grammatiken

Sei $G = (T, V, S, P)$ eine Grammatik. G heißt ohne weitere Einschränkungen **allgemein**.

Allgemeine Grammatiken heißen auch **Chomsky Typ-0 Grammatiken**.

Allgemeine Grammatiken

Allgemeine Grammatiken erzeugen genau die …

- … semi-entscheidbaren …
- … von Turingmaschinen akzeptierten …

… Sprachen.

Chomsky-Hirachie

Typ-3 Reguläre Sprachen

Typ-2 Kontextfreie Sprachen

Typ-1 Kontextsensitive Sprachen

Typ-0 Rekursiv-aufzählbare Sprachen