Was bisher geschah

Sprache L ist vom Chomsky-Typ i gdw.
eine Grammatik $G = (N, T, P, S)$ mit $L(G) = L \setminus \{\varepsilon\}$ existiert.
Chomsky-Hierarchie: G ist vom Chomsky-Typ

0 immer,

1 , $\forall(l \to r) \in P : |l| \leq |r|$ (monoton, kontextsensitiv)

2 , $\forall(l \to r) \in P : |l| \leq |r| \wedge l \in N$ (kontextfrei)

3 , $\forall(l \to r) \in P : |l| \leq |r| \wedge l \in N \wedge r \in (T \cup (T \circ N))$ (regulär)

reguläre Sprachen ($\mathcal{L}_3 = \mathrm{REG} = \mathrm{REC(NFA)} = \mathrm{REC(DFA)}$):

- endliche Repräsentation durch
  - Darstellung als reguläre Ausdrücke
  - Erzeugung durch reguläre Grammatiken,
  - Akzeptanz durch NFA (DFA, ε-NFA, ...),
- abgeschlossen unter $\cup, \cap, \overline{}, \circ, {}^{*}, {}^{R}$

Fragen für kontextfreie Sprachen

Wiederholung:
Eine Sprache $L \subseteq X^*$ heißt genau dann kontextfrei, wenn für eine kontextfreie Grammatik (Chomsky-Typ 2) G gilt $L \setminus \{\varepsilon\} = L(G)$.

CF ($= \mathcal{L}_2$): Menge aller kontextfreien Sprachen

- Unter welchen der Operationen $\cup, \cap, \overline{}, \circ, {}^*, {}^R$ ist CF abgeschlossen?
- Gibt es ein passendes (deterministisches, eindeutiges) Maschinenmodell für kontextfreie Sprachen?
- Existieren algorithmische Lösungsverfahren für folgende Probleme für kontextfreie Sprachen:
  - Wortproblem
  - Leerheit, Vollständigkeit, Endlichkeit
  - Inklusion, Gleichheit, Disjunktheit

Kontextfreie Grammatiken – Wiederholung

Grammatik $G = (N, T, P, S)$ vom Chomsky-Typ 2

Alle Regeln in P haben die Form $l \to r$ mit
$l \in N$ und $r \in (N \cup T)^+$

w in G ableitbar gdw. $S \to_P^* w$

Grammatik $G = (N, T, P, S)$ erzeugt die Sprache
$L(G) = \{w \in T^* \mid S \to_P^* w\}$

Beispiel: Grammatik $G = (\{S\}, \{a, b\}, \{S \to aSb|ab\}, S)$

- ist vom Chomsky-Typ 2
- Ableitung für $aaabbb$ in G: $S \to aSb \to aaSbb \to aaabbb$
- G erzeugt die Sprache $L(G) = \{a^n b^n \mid n > 0\}$
  (vom Chomsky-Typ 2)

Für $L = \{a^n b^n \mid n > 0\}$ gilt $L \in \text{CF} \setminus \text{REG}$, d.h.

- L ist kontextfrei.
- L ist nicht regulär. (Wie kann man das zeigen?)

WH: Schubfachschluss-Prinzip

Schubfachschluss-Prinzip (Pigeonhole principle):
Für $|O| > |S|$ existiert keine injektive Funktion $f : O \to S$.

$$\forall O \forall S \forall f : ((|O| > |S| \land f : O \to S) \to \exists x \exists y : (x \neq y \land f(x) = f(y)))$$

Anschaulich:
Verteilt man mehr als n Objekte in n Schubfächer, dann gibt es (wenigstens) ein Schubfach, welches (wenigstens) zwei Objekte enthält.
Beispiele:

- Von 13 Personen haben mindestens zwei im selben Monat Geburtstag.
- Wieviele Karten aus einem Skatblatt muss man ziehen, damit zwei derselben Farbe dabei sind?
- vier verschiedenfarbige Paare Socken im Dunklen:
  Wieviele Socken muss man (blind) nehmen, damit ein vollständiges Paar dabei ist?
- Bei jeder Wanderung mit $> n$ Pausen in einem Gebiet mit n Rastplätzen waren wenigsten zwei Pausen am selben Platz.

Nichtreguläre Sprachen

Fakt

Die Sprache $L = \{a^n b^n \mid n > 0\}$ *ist nicht NFA-akzeptierbar.*

WH Definition REC(NFA): $L \in \text{REC(NFA)}$ gdw. $\exists$ NFA $A : (L(A) = L)$

Vorbereitung für Beweis (indirekt):

$$\begin{aligned}\forall L \subseteq X^* : L \notin \text{REC(NFA)} \quad &\text{gdw.} \quad \neg\exists A : (L(A) = L) \text{ gdw. } \forall A : \neg(L(A) = L) \\ &\text{gdw.} \quad \forall A : (\ \underbrace{(L(A) = L)}_{\text{Annahme } P(A)} \to \underbrace{\neg(L(A) = L)}_{\text{Folgerung } \neg P(A)}\ )\end{aligned}$$

Beweis: Annahme: für NFA $A = (X, Q, \delta, I, F)$ mit $|Q| = k$ gilt $L(A) = L$

Wegen $a^k b^k \in L$ existiert ein akzeptierender Pfad für $a^k b^k$ in A:

$$q_0 \xrightarrow{a} q_1 \xrightarrow{a} \cdots \xrightarrow{a} q_k \xrightarrow{b} q_{k+1} \xrightarrow{b} \cdots \xrightarrow{b} q_{2k} \quad \text{mit } q_0 \in I, q_{2k} \in F$$

SFS: O: Positionen im a^k-Teil $|\{0, \ldots, k\}| > k$, S: Zustände $|Q| = k$

Nach SFS existieren $i, j \in \{0, \ldots, k\}$ mit $i < j$ und $q_i = q_j$.

$$q_0 \xrightarrow{a} q_1 \xrightarrow{a} \cdots \xrightarrow{a} q_i = q_j \xrightarrow{a} q_{j+1} \xrightarrow{a} \cdots \xrightarrow{a} q_k \xrightarrow{b} q_{k+1} \xrightarrow{b} \cdots \xrightarrow{b} q_{2k}$$

ist also akzeptierender Pfad für $a^{k-(j-i)} b^k$ in A, d.h $a^{k-(j-i)} b^k \in L(A)$

Aus $a^{k-(j-i)} b^k \notin L$ folgt $\neg(L(A) = L)$ im Widerspruch zur Annahme.

Also ist die Annahme falsch, d.h. es existiert kein NFA A mit $L = L(A)$.

Weitere nichtreguläre Sprachen

hilfreiches Prinzip:
Da REC(NFA) unter $\cap$ abgeschlossen ist, d.h. $\forall L, L', L'' \subseteq X^*$

$$(\ L \cap L' = L'' \land L \in \text{REC(NFA)} \land L' \in \text{REC(NFA)} \ ) \to L'' \in \text{REC(NFA)}$$

gilt für alle Sprachen L, L', L'': (ÜA)

$$(\ L \cap L' = L'' \land L' \in \text{REC(NFA)} \land L'' \notin \text{REC(NFA)} \ ) \to L \notin \text{REC(NFA)}$$

- $L = \{w \in \{a,b\}^* \mid |w|_a = |w|_b > 0\} \notin \text{REC(NFA)}$
  (wegen $L \cap \underbrace{L(a^*b^*)}_{\in \text{REC(NFA)}} = \{a^n b^n \mid n > 0\} \notin \text{REC(NFA)}$)
- $L = \{w \in \{a,b\}^* \mid 2|w|_a = |w|_b \in \mathbb{N}\} \notin \text{REC(NFA)}$
  (wegen $L \cap L(a^*b^*) = \{a^n b^{2n} \mid n > 0\} \notin \text{REC(NFA)}$
  nach Schubfachschluss).
- $L = \{w \in \{a,b\}^* \mid w = w^R\} \notin \text{REC(NFA)}$ (Palindrome)
  (wegen $L \cap L(a^*ba^*) = \{a^n ba^n \mid n \in \mathbb{N}\} \notin \text{REC(NFA)}$)
- $L = \{a^{(n^2)} \mid n \in \mathbb{N}\} \notin \text{REC(NFA)}$
  (nach Schubfachschluss, Wortlängen)

Ableitungsbäume für kontextfreie Grammatiken

Beispiel: Grammatik $G = (N, T, P, E)$ mit $N = \{E, F\}$,
$T = \{a, b, c, (,), +, *\}$,

$$P = \{E \to (E + E), E \to F * F, F \to E, E \to a, E \to b, E \to c\}$$

Ableitung für $w = (c * a + (b + a * a))$

allgemein: Grammatik $G = (N, T, P, S)$

Ableitung $S \to_P w_1 \to_P w_2 \to_P \cdots \to_P w_n = w$ für w in G

Ableitungsbaum zu einem Wort w in der kontextfreien Grammatik G (induktive Definition):

- Wurzel mit Markierung S
- für jeder Anwendung einer Regel $A \to_P r_1 \cdots r_n$: Verzweigung bei Symbol A in n Kinder mit Markierungen $r_1 \ldots, r_n$
- Markierung der Blätter (von links nach rechts): Wort w

Ein Ableitungsbaum repräsentiert i.A. mehrere Ableitungen.

Rechts- und Linksableitungen

Eine Ableitung für w in G heißt

Linksableitung , falls in jedem Schritt das am weitesten links stehende Nichtterminal

Rechtsableitung , falls in jedem Schritt das am weitesten rechts stehende Nichtterminal

ersetzt wird.

Zu jedem Ableitungsbaum für w in G existieren

- genau eine Linksableitung für w in G und
- genau eine Rechtsableitung für w in G.

Abschlusseigenschaften von CF

Satz

Sind L_1 und L_2 kontextfreie Sprachen, dann sind auch kontextfrei:

- $L_1 \cup L_2$
- $L_1 \circ L_2$
- L_1^*
- L_1^R

Konstruktionen für $L_i = L(G_i)$ mit $G_i = (N_i, T_i, P_i, S_i)$ für $i \in \{1, 2\}$,
wobei $N_1 \cap N_2 = \emptyset$ (falls nötig, umbenennen)
neues Nichtterminal $S' \notin N_1 \cup N_2$

$$
\begin{aligned}
L_1 \cup L_2 &= L(G_\cup) \text{ mit } G_\cup = (N_1 \cup N_2 \cup \{S'\}, T_1 \cup T_2, P_\cup, S'),\\
&\quad \text{wobei } P_\cup = P_1 \cup P_2 \cup \{S' \to S_1, S' \to S_2\}\\
L_1 \circ L_2 &= L(G_\circ) \text{ mit } G_\circ = (N_1 \cup N_2 \cup \{S'\}, T_1 \cup T_2, P_\circ, S'),\\
&\quad \text{wobei } P_\circ = P_1 \cup P_2 \cup \{S' \to S_1 S_2\}\\
L_1^* &= L(G_*) \text{ mit } G_* = (N_1 \cup \{S'\}, T_1, P_*, S') \text{, wobei}\\
&\quad P_* = P_1 \cup \{S' \to \varepsilon, S' \to S_1 S'\}\\
L_1^R &\ \text{(war ÜA 3.4)}
\end{aligned}
$$

noch offen: CF abgeschlossen unter $^-, \cap$?

Mehrdeutige kontextfreie Grammatiken

Kontextfreie Grammatik G heißt

eindeutig, falls für jedes Wort $w \in L(G)$ genau ein Ableitungsbaum in G existiert.

mehrdeutig, sonst.

Beispiel: Ableitungsbäume für $w = aaa$ in der Grammatik $G = (N, T, P, E)$ mit $N = \{S\}$, $T = \{a\}$, $P = \{S \to SS, S \to a\}$

Eindeutigkeit von Grammatiken ist wichtig beim Entwurf von Programmiersprachen (Syntax)

„dangling else"-Problem:

```
<Anw> ::= if <Ausdr> then <Anw>
<Anw> ::= if <Ausdr> then <Anw> else <Anw>
```

Was bedeutet:

```
if C1 then if C2 then A else B
```

(Anwendungen im Modul Prinzipien von Programmiersprachen)

Inhärent mehrdeutige Sprachen

Sprachen L, für die keine eindeutige Grammatik G mit $L = L(G)$ existieren, heißen inhärent mehrdeutig.

Fakt

Die kontextfreie Sprache
$L = \{a^l b^m c^n \mid l, m, n > 0 \wedge (l = m \vee m = n)\}$
ist inhärent mehrdeutig.