Teil V

Weiterführende Themen, Teil 2: Modelle mit Wiederholungsoperatoren

Definition

- Seien Σ und X disjunkte Alphabete (**Terminalalphabet** und **Variablenalphabet**).
- **Pattern**: nicht-leeres Wort $\alpha \in (\Sigma \cup X)^{+}$

Pattern und ihre Sprachen

Definition

- Seien Σ und X disjunkte Alphabete (**Terminalalphabet** und **Variablenalphabet**).
- **Pattern**: nicht-leeres Wort $\alpha \in (\Sigma \cup X)^+$
- Σ^+-**Ersetzung:**
  Homomorphismus $\sigma\colon (\Sigma \cup X)^+ \to \Sigma^+$ mit:
  - $\sigma(a) = a$ für alle $a \in \Sigma$,
  - $\sigma(x) \neq \varepsilon$ für alle $x \in X$.

Pattern und ihre Sprachen

Definition

- Seien Σ und X disjunkte Alphabete (**Terminalalphabet** und **Variablenalphabet**).
- **Pattern**: nicht-leeres Wort $\alpha \in (\Sigma \cup X)^+$
- Σ^+-**Ersetzung:**
  Homomorphismus $\sigma\colon (\Sigma \cup X)^+ \to \Sigma^+$ mit:
  - $\sigma(a) = a$ für alle $a \in \Sigma$,
  - $\sigma(x) \neq \varepsilon$ für alle $x \in X$.

$\mathcal{L}_\Sigma(\alpha) := \{\sigma(\alpha) \mid \sigma$ ist eine Σ^+-Ersetzung$\}$.

Pattern und ihre Sprachen

Definition

- Seien Σ und X disjunkte Alphabete (**Terminalalphabet** und **Variablenalphabet**).
- **Pattern**: nicht-leeres Wort $\alpha \in (\Sigma \cup X)^+$
- Σ^+-**Ersetzung:**
  Homomorphismus $\sigma \colon (\Sigma \cup X)^+ \to \Sigma^+$ mit:
  - $\sigma(a) = a$ für alle $a \in \Sigma$,
  - $\sigma(x) \neq \varepsilon$ für alle $x \in X$.

$\mathcal{L}_\Sigma(\alpha) := \{\sigma(\alpha) \mid \sigma$ ist eine Σ^+-Ersetzung$\}$.

- L ist eine Patternsprache, wenn $L = \mathcal{L}(\alpha)$ für ein Pattern α.

Pattern und ihre Sprachen

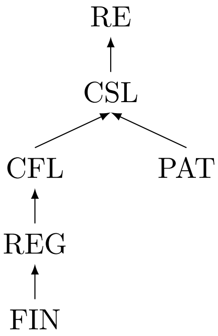
Definition

- Seien Σ und X disjunkte Alphabete (**Terminalalphabet** und **Variablenalphabet**).
- **Pattern**: nicht-leeres Wort $\alpha \in (\Sigma \cup X)^+$
- Σ^+-**Ersetzung:**
  Homomorphismus $\sigma\colon (\Sigma \cup X)^+ \to \Sigma^+$ mit:
  - $\sigma(a) = a$ für alle $a \in \Sigma$,
  - $\sigma(x) \neq \varepsilon$ für alle $x \in X$.

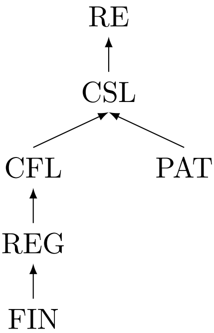
$\mathcal{L}_\Sigma(\alpha) := \{\sigma(\alpha) \mid \sigma \text{ ist eine } \Sigma^+\text{-Ersetzung}\}$.

- L ist eine Patternsprache, wenn $L = \mathcal{L}(\alpha)$ für ein Pattern α.
- PAT ist die Klasse aller Patternsprachen.

PAT und die Chomsky-Hierarchie



PAT und die Chomsky-Hierarchie



Lemma

Sei $L \in \mathrm{PAT}$. Dann gilt entweder $|L| = 1$ oder $|L| = \infty$.

Das Wortproblem für Pattern

Wortproblem für Pattern

Eingabe: Pattern $\alpha \in (\Sigma \cup X)^+$, $w \in \Sigma^+$
Frage: Ist $w \in \mathcal{L}_\Sigma(\alpha)$?

Wortproblem für Pattern

Eingabe: Pattern $\alpha \in (\Sigma \cup X)^+$, $w \in \Sigma^+$
Frage: Ist $w \in \mathcal{L}_\Sigma(\alpha)$?

Satz

Das Wortproblem für Pattern ist NP-vollständig.

Das Wortproblem für Pattern

Wortproblem für Pattern

Eingabe: Pattern $\alpha \in (\Sigma \cup X)^+$, $w \in \Sigma^+$
Frage: Ist $w \in \mathcal{L}_\Sigma(\alpha)$?

Satz

Das Wortproblem für Pattern ist NP-vollständig.

Idee

- Reduktion von 3-Färbbarkeit.
- Konstruiere Σ, α, w, so dass $w \in \mathcal{L}_\Sigma(\alpha)$ gdw. G 3-färbbar.

Inklusion von Patternsprachen

Definition

- Seien $\alpha, \beta \in (\Sigma \cup X)^+$ Pattern.
- $\alpha \preceq \beta$ (α aus β erzeugbar):
  Es existiert ein Homomorphismus
  $h\colon (\Sigma \cup X)^+ \to (\Sigma \cup X)^+$ mit
  1. $h(a) = a$ für alle $a \in \Sigma$,
  2. $h(x) \neq \varepsilon$ für alle $x \in X$,
  3. $h(\beta) = \alpha$.

Inklusion von Patternsprachen

Definition

- Seien $\alpha, \beta \in (\Sigma \cup X)^+$ Pattern.
- $\alpha \preceq \beta$ (α aus β erzeugbar):
  Es existiert ein Homomorphismus
  $h\colon (\Sigma \cup X)^+ \to (\Sigma \cup X)^+$ mit
  1. $h(a) = a$ für alle $a \in \Sigma$,
  2. $h(x) \neq \varepsilon$ für alle $x \in X$,
  3. $h(\beta) = \alpha$.

Lemma

Aus $\alpha \preceq \beta$ folgt $\mathcal{L}_\Sigma(\alpha) \subseteq \mathcal{L}_\Sigma(\beta)$.

Inklusion von Patternsprachen

Definition

- Seien $\alpha, \beta \in (\Sigma \cup X)^+$ Pattern.
- $\alpha \preceq \beta$ (α aus β erzeugbar):
  Es existiert ein Homomorphismus $h\colon (\Sigma \cup X)^+ \to (\Sigma \cup X)^+$ mit
  1. $h(a) = a$ für alle $a \in \Sigma$,
  2. $h(x) \neq \varepsilon$ für alle $x \in X$,
  3. $h(\beta) = \alpha$.

$$\begin{tikzcd} & w \\ \alpha \arrow[ur, "\sigma"] & \\ & \beta \arrow[ul, "h"] \arrow[uu, "\sigma \circ h"'] \end{tikzcd}$$

Lemma

Aus $\alpha \preceq \beta$ folgt $\mathcal{L}_\Sigma(\alpha) \subseteq \mathcal{L}_\Sigma(\beta)$.

Inklusion von Patternsprachen

Definition

- Seien $\alpha, \beta \in (\Sigma \cup X)^+$ Pattern.
- $\alpha \preceq \beta$ (α aus β erzeugbar):
  Es existiert ein Homomorphismus
  $h\colon (\Sigma \cup X)^+ \to (\Sigma \cup X)^+$ mit
  1. $h(a) = a$ für alle $a \in \Sigma$,
  2. $h(x) \neq \varepsilon$ für alle $x \in X$,
  3. $h(\beta) = \alpha$.

$$
\begin{tikzcd}
 & w \\
\alpha \arrow[ur, "\sigma"] & \\
 & \beta \arrow[ul, "h"] \arrow[uu, "\sigma \circ h"']
\end{tikzcd}
$$

Vorsicht!

Andere Richtung gilt nicht.

Lemma

Aus $\alpha \preceq \beta$ folgt $\mathcal{L}_\Sigma(\alpha) \subseteq \mathcal{L}_\Sigma(\beta)$.

Reguläre Ausdrücke in der Praxis

- **Eckige Klammern:** `[a-z]`
- **Zeichenklassen:** `[:alpha:]`, `[:digit:]`
- **Fragezeichen:** `a?`
- **Punkt:** `.`
- **Quantoren:** `a{2,7}`

Reguläre Ausdrücke in der Praxis

- **Eckige Klammern:** `[a-z]`
- **Zeichenklassen:** `[:alpha:]`, `[:digit:]`
- **Fragezeichen:** `a?`
- **Punkt:** `.`
- **Quantoren:** `a{2,7}`

PERL

`((a|b)*)\1`

Reguläre Ausdrücke in der Praxis

- **Eckige Klammern:** `[a-z]`
- **Zeichenklassen:** `[:alpha:]`, `[:digit:]`
- **Fragezeichen:** `a?`
- **Punkt:** `.`
- **Quantoren:** `a{2,7}`

PERL

`((a|b)*)\1`

Python

`(?P<X>(a|b)*)(?P=X)`

Erweiterte reguläre Ausdrücke

Definition

Erweiterte reguläre Ausdrücke:

- wie reguläre Ausdrücke, aber zusätzlich Variablenalphabet X

Erweiterte reguläre Ausdrücke

Definition

Erweiterte reguläre Ausdrücke:

- wie reguläre Ausdrücke, aber zusätzlich Variablenalphabet X
- **Variablenbindung:** $\langle x\colon \alpha \rangle$

Erweiterte reguläre Ausdrücke

Definition

Erweiterte reguläre Ausdrücke:

- wie reguläre Ausdrücke, aber zusätzlich Variablenalphabet X
- **Variablenbindung:** $\langle x \colon \alpha \rangle$
- **Variablenaufruf:** x

Erweiterte reguläre Ausdrücke

Definition

Erweiterte reguläre Ausdrücke:

- wie reguläre Ausdrücke, aber zusätzlich Variablenalphabet X
- **Variablenbindung:** $\langle x\colon \alpha\rangle$
- **Variablenaufruf:** x

- $\Sigma := \{\mathtt{a}, \mathtt{b}\}$
- $\alpha := \langle x\colon (\mathtt{a} \mid \mathtt{b})^*\rangle\, x$

Erweiterte reguläre Ausdrücke

Definition

Erweiterte reguläre Ausdrücke:

- wie reguläre Ausdrücke, aber zusätzlich Variablenalphabet X
- **Variablenbindung:** $\langle x \colon \alpha \rangle$
- **Variablenaufruf:** x

- $\Sigma := \{\mathtt{a}, \mathtt{b}\}$
- $\alpha := \langle x \colon (\mathtt{a} \mid \mathtt{b})^* \rangle \, x$
- $\mathcal{L}(\alpha) = \mathrm{COPY}_\Sigma$

Erweiterte reguläre Ausdrücke

Definition

Erweiterte reguläre Ausdrücke:

- wie reguläre Ausdrücke, aber zusätzlich Variablenalphabet X
- **Variablenbindung:** $\langle x \colon \alpha \rangle$
- **Variablenaufruf:** x

- $\Sigma := \{\mathtt{a}, \mathtt{b}\}$
- $\alpha := \langle x \colon (\mathtt{a} \mid \mathtt{b})^* \rangle \, x$
- $\mathcal{L}(\alpha) = \mathrm{COPY}_\Sigma$

- **erweitert reguläre Sprache:** wird von erweitertem regulären Ausdruck erzeugt
- XREG: Klasse aller erweitert regulären Sprachen.

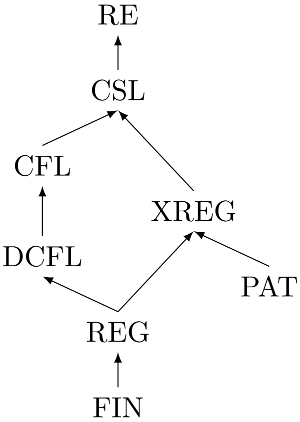
Noch ein Pumping-Lemma! Juhu!

Pumping-Lemma für XREG

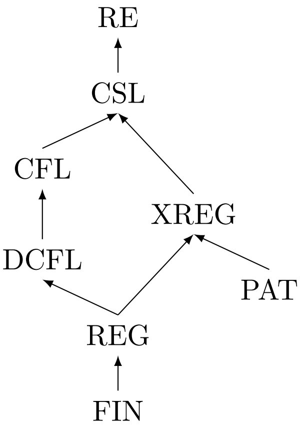
Sei Σ ein Alphabet.
Für jedes $L \in \mathrm{XREG}_\Sigma$ existiert eine **Pumpkonstante** n_L, so dass für jedes Wort $z \in L$ mit $|z| \geq n_L$ gilt: Es existieren ein $m \in \mathbb{N}_{>0}$ und Wörter $x_0, x_1, \ldots, x_m, y \in \Sigma^*$ mit

1. $x_0 \cdot y \cdot x_1 \cdot y \cdot x_2 \cdots x_{m-1} \cdot y \cdot x_m = z$,
2. $|x_0 y| \leq n_L$,
3. $|y| \geq 1$,
4. $x_0 \cdot y^i \cdot x_1 \cdot y^i \cdot x_2 \cdots x_{m-1} \cdot y^i \cdot x_m \in L$ für alle $i \in \mathbb{N}$.

XREG im Vergleich



XREG im Vergleich



Satz

Das Wortproblem für erweiterte reguläre Ausdrücke ist NP-vollständig.