

# $\varepsilon$ -pravidla

**Definice 3.13.** Řekneme, že CFG  $\mathcal{G} = (N, \Sigma, P, S)$  je **bez  $\varepsilon$ -pravidel** právě když buď

1.  $P$  neobsahuje žádné  $\varepsilon$ -pravidlo (tj. pravidlo tvaru  $A \rightarrow \varepsilon$ ) nebo
2. v  $P$  existuje právě jedno  $\varepsilon$ -pravidlo  $S \rightarrow \varepsilon$  a  $S$  se nevyskytuje na pravé straně žádného pravidla z  $P$ .

# Příklad

$\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$ , kde  $P$  obsahuje pravidla

$$S \rightarrow aAbBc$$

$$A \rightarrow BB \mid a \mid \varepsilon$$

$$B \rightarrow AcA \mid b$$

# Algoritmus pro odstranění $\varepsilon$ -pravidel

**Vstup:** CFG  $\mathcal{G} = (N, \Sigma, P, S)$

**Výstup:** CFG  $\mathcal{G}' = (N', \Sigma, P', S')$  bez  $\varepsilon$ -pravidel splňující  $L(\mathcal{G})=L(\mathcal{G}')$

```
1 Zkonstruuuj  $N_\varepsilon = \{A \in N \mid A \Rightarrow^* \varepsilon\}$ 
2 Množinu pravidel  $P'$  zkonstruuuj takto:
3 foreach  $A \rightarrow X_1 \dots X_n \in P$  do
4   přidej do  $P'$  všechna pravidla tvaru  $A \rightarrow \alpha_1 \dots \alpha_n$  splňující
5     (a) pokud  $X_i \notin N_\varepsilon$  pak  $\alpha_i = X_i$ 
6     (b) pokud  $X_i \in N_\varepsilon$  pak  $\alpha_i$  je buď  $X_i$ , nebo  $\varepsilon$ 
7     (c) ne všechna  $\alpha_i$  jsou  $\varepsilon$ 
8 od
9 if  $S \in N_\varepsilon$  then přidej do  $P'$  pravidla  $S' \rightarrow S \mid \varepsilon$  ( $S' \notin N \cup \Sigma$ );
10      $N' := N \cup \{S'\}$ 
11   else  $N' := N$ ;  $S' := S$  fi
```

# Příklad

$\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$ , kde  $P$  obsahuje pravidla

$$S \rightarrow aAbBc \mid AB$$

$$A \rightarrow BB \mid a \mid \varepsilon$$

$$B \rightarrow AA \mid b$$

# Korektnost algoritmu

**Konečnost.**

**Výsledná gramatika je bez  $\varepsilon$ -pravidel.**

**Ekvivalence gramatik.**

# Jednoduchá pravidla

**Jednoduchým pravidlem** nazýváme každé pravidlo tvaru  $A \rightarrow B$ , kde  $A, B \in N$ .

$$S \rightarrow aAbBc$$

$$A \rightarrow aA \mid B \mid a$$

$$B \rightarrow bB \mid b$$

# Algoritmus pro odstranění jednoduchých pravidel

**Vstup:** CFG  $\mathcal{G} = (N, \Sigma, P, S)$  bez  $\varepsilon$ -pravidel

**Výstup:** CFG  $\mathcal{G}' = (N, \Sigma, P', S)$  bez jednoduchých a  $\varepsilon$ -pravidel, kde  
 $L(\mathcal{G}) = L(\mathcal{G}')$

```
1 foreach  $A \in N$  do
2    $i := 0$ ;  $N_i := \{A\}$ 
3   repeat  $i := i + 1$ 
4      $N_i := N_{i-1} \cup \{C \mid B \rightarrow C \in P, B \in N_{i-1}\}$ 
5   until  $N_i = N_{i-1}$ 
6    $N_A := N_i$ 
7 od
8  $P' := \emptyset$ 
9 foreach  $A \in N$  do
10   $P' := P' \cup \{A \rightarrow \alpha \mid B \in N_A \wedge B \rightarrow \alpha \in P \text{ není jednoduché}\}$ 
11 od
```

# Příklad

$\mathcal{G} = (\{S, A, B, C\}, \{a, b, c\}, P, S)$ ,

kde  $P$  obsahuje pravidla

$S \rightarrow ABC$

$A \rightarrow aA \mid B \mid a$

$B \rightarrow bB \mid A$

$C \rightarrow cC \mid A$



# Korektnost algoritmu

**Konečnost.**

**Výsledná gramatika neobsahuje jednoduchá pravidla.**

**Ekvivalence gramatik:**

$L(\mathcal{G}') \subseteq L(\mathcal{G})$  Nechť  $w \in L(\mathcal{G}')$ , pak existuje derivace

$$S = \alpha_0 \Rightarrow_{\mathcal{G}'} \alpha_1 \Rightarrow_{\mathcal{G}'} \dots \Rightarrow_{\mathcal{G}'} \alpha_n = w.$$

Pokud bylo při kroku  $\alpha_i \Rightarrow_{\mathcal{G}'} \alpha_{i+1}$  použito pravidlo  $A \rightarrow \beta$ , pak existuje nějaké  $B \in N_A$  takové, že v  $\mathcal{G}$  platí  $A \Rightarrow^* B$  a  $B \Rightarrow \beta$ .

Tedy v  $\mathcal{G}$  platí  $A \Rightarrow^* \beta$  a  $\alpha_i \Rightarrow^* \alpha_{i+1}$ .

$L(\mathcal{G}) \subseteq L(\mathcal{G}')$  Nechť  $w \in L(\mathcal{G})$ , pak existuje levá derivace

$$S = \alpha_0 \Rightarrow_{\mathcal{G}} \alpha_1 \Rightarrow_{\mathcal{G}} \dots \Rightarrow_{\mathcal{G}} \alpha_n = w.$$

Tu lze rozdělit na úseky tak, že v celém úseku se použila pouze jednoduchá pravidla anebo žádné jednoduché pravidlo. Úseky s jednoduchými pravidly lze nahradit.

# Vlastní bezkontextová gramatika

**Definice 3.17.** CFG  $\mathcal{G} = (N, \Sigma, P, S)$  se nazývá **necyklická**, právě když neexistuje  $A \in N$  takový, že  $A \Rightarrow^+ A$ .

$\mathcal{G}$  se nazývá **vlastní**, právě když je bez nepoužitelných symbolů, bez  $\varepsilon$ -pravidel a necyklická.

**Věta 3.18.** Ke každému neprázdnému bezkontextovému jazyku existuje **vlastní** bezkontextová gramatika, která jej generuje.

**Důkaz.** Z bezkontextové gramatiky pro neprázdný jazyk odstraníme  $\varepsilon$ -pravidla a jednoduchá pravidla. Odstraněním nepoužitelných symbolů pak získáme vlastní gramatiku.  $\square$

# Chomského normální forma

**Definice 3.19.** Bezkontextová gramatika  $\mathcal{G} = (N, \Sigma, P, S)$  je v **Chomského normální formě (CNF)**  $\stackrel{def}{\iff}$   $\mathcal{G}$  je bez  $\varepsilon$ -pravidel a každé pravidlo z  $P$  má jeden z těchto tvarů:

1.  $A \rightarrow BC$ , kde  $B, C \in N$
2.  $A \rightarrow a$ , kde  $a \in \Sigma$
3.  $S \rightarrow \varepsilon$

**Věta 3.21.** Každý bezkontextový jazyk lze generovat bezkontextovou gramatikou v Chomského normální formě.

# Příklad

$\mathcal{G} = (\{S, A, B\}, \{a, b\}, P, S)$ ,  
kde  $P$  obsahuje pravidla

$$S \rightarrow AS \mid a$$

$$A \rightarrow AB \mid AA \mid a$$

$$B \rightarrow b$$

# Lemma o substituci

## Lemma 3.20. (o substituci)

Nechť  $\mathcal{G} = (N, \Sigma, P, S)$  je CFG. Nechť  $A \rightarrow \alpha_1 B \alpha_2 \in P$ .

Nechť  $B \rightarrow \beta_1 \mid \dots \mid \beta_r$  jsou všechna pravidla v  $P$  tvaru  $B \rightarrow \alpha$ .

Definujme  $\mathcal{G}' = (N, \Sigma, P', S)$ , kde

$$P' = (P \setminus \{A \rightarrow \alpha_1 B \alpha_2\}) \cup \{A \rightarrow \alpha_1 \beta_1 \alpha_2 \mid \dots \mid \alpha_1 \beta_r \alpha_2\}.$$

Pak  $L(\mathcal{G}) = L(\mathcal{G}')$ .

# Algoritmus transformace do CNF

1.  $L = \emptyset$

2.  $L \neq \emptyset$

Gramatiku pro  $L$  převedeme na vlastní a bez jednoduchých pravidel.

$$X \rightarrow \varepsilon$$

$$X \rightarrow a$$

$$X \rightarrow A$$

$$X \rightarrow ab$$

$$X \rightarrow aB$$

$$X \rightarrow Ab$$

$$X \rightarrow AB$$

⋮

$$X \rightarrow aBcD$$

# Lemma o vkládání pro bezkontextové jazyky

**Věta 3.24.** Nechť  $L$  je CFL. Pak existují  $p, q \in \mathbb{N}$  (závisející na  $L$ ) taková, že každé slovo  $z \in L$  delší než  $p$  lze psát ve tvaru  $z = uvwxy$ , kde

- alespoň jedno ze slov  $v, x$  je neprázdné (tj.  $vx \neq \varepsilon$ ),
- $|vwx| \leq q$  a
- $uv^iwx^iy \in L$  pro každé  $i \in \mathbb{N}_0$ .

**Poznámka 3.25.** Tvrzení zůstává v platnosti i když namísto konstant  $p, q$  budeme všude psát jen (jedinou) konstantu  $n$ .

# Důkaz Lemmatu o vkládání pro bezkontextové jazyky

Nechť  $L$  je generován gramatikou v CNF.

**délka cesty** z kořene do listu

= počet modrých hran

= počet neterminálů na cestě - 1

**hloubka stromu**

= maximální délka cesty

Derivační strom hloubky  $k$  má max.  $2^k$  listů  $\implies$  slovo délky nejvýše  $2^k$ .

Derivační strom pro slovo delší než  $2^{k-1}$  má cestu délky alespoň  $k$ .

Tato cesta obsahuje alespoň  $k + 1$  neterminálů.



# Důkaz Lemmatu o vkládání pro bezkontextové jazyky

Nechť  $L$  generován gramatikou  $\mathcal{G} = (N, \Sigma, P, S)$ , která je v CNF.

Označme  $k = \text{card}(N)$  a položme  $p = 2^{k-1}$ ,  $q = 2^k$ .

Nechť  $z \in L$  je slovo delší než  $p$ . Pak v libovolném derivačním stromu slova  $z$  existuje cesta délky alespoň  $k$ . Zvolme pevně jeden takový strom  $T$  a v něm (libovolnou) nejdelší cestu  $C$ .

Na cestě  $C$  lze zvolit tři uzly  $u_1, u_2, u_3$  s vlastnostmi:

1. uzly  $u_1, u_2$  jsou označeny týmž neterminálem, řekněme  $A$ ,
2.  $u_1$  leží blíže ke kořenu než  $u_2$ ,
3.  $u_3$  je list a
4. cesta z  $u_1$  do  $u_3$  má délku nejvýše  $k$ .



# Použití Lemmatu o vkládání pro bezkontextové jazyky

Lemma o vkládání je implikace  $P \implies Q$ , kde  $P$  je výrok, že  $L$  je CFL a  $Q$  jsou uvedené vlastnosti.

Obměnu Lemmatu o vkládání  $\neg Q \implies \neg P$  lze použít k důkazu, že nějaký jazyk  $L$  **není** CFL — stačí, když ukážeme platnost  $\neg Q$ .

$\neg Q$ :

1. Pro libovolnou konstantu  $n \in \mathbb{N}$
2. existuje slovo  $z \in L$  delší než  $n$  takové, že
3. pro všechny slova  $u, v, w, x, y$  splňující  
 $z = uvwxy$ ,  $vx \neq \varepsilon$  a  $|vwx| \leq n$
4. existuje  $i \in \mathbb{N}_0$  takové, že  $uv^iwx^iy \notin L$ .

# Příklad použití Lemmatu o vkládání

$$L = \{a^i b^i c^i \mid i \geq 1\}$$

1. Pro libovolnou konstantu  $n \in \mathbb{N}$
2. existuje slovo  $z \in L$  delší než  $n$  takové, že
3. pro všechny slova  $u, v, w, x, y$  splňující  
 $z = uvwxy$ ,  $vx \neq \varepsilon$  a  $|vwx| \leq n$
4. existuje  $i \in \mathbb{N}_0$  takové, že  $uv^iwx^iy \notin L$ .

$\implies L$  není CFL.