Přijímající mód uspořádaných gramatik

Stanislav Židek

10. srpna 2010

Obsah

Abstrakt

Tato práce se zabývá přijímajícími gramatikami. Ty tvoří logický protějšek obvyklého chápání formálních gramatik jako čistě generujících struktur.

V některých případech je přijímající mód právě tak silný jako mód generující, nicméně existují i případy, kdy je přijímající mód striktně silnější. Toto je na první pohled neočekávaný výsledek, a proto se v tomto textu zaměříme na popis a důkaz jednoho konkrétního případu, kdy k tomuto jevu dojde, a sice uspořádaných gramatik.

Kapitola 1

Úvod

Na každý mechanismus, který generuje určitou množinu slov, a tedy charakterizuje nějaký formální jazyk, se zároveň můžeme dívat jako na mechanismus, který slova přijímá. Tyto dva úhly pohledu sice nemají přílišný význam u nejznámějších typů formálních gramatik, u Chomského hierarchie, neboť v obou případech gramatiky charakterizují stejnou třídu jazyků, nicméně tento trend není obecně platný a lze nalézt případy, jež nás překvapí změnou síly při použití přijímající verze téhož mechanismu.

Většina typů gramatik a systémů byla studována téměř výhradně pouze z generující perspektivy. Především jde o gramatiky s řízeným přepisováním (důkladně studovány například v [3]) a o gramatiky/systémy umožňující určitou míru paralelismu. To je poměrně překvapivé zjištění, vezmeme-li v úvahu, že praktické využití formálních jazyků spočívá ve většině případů ve využití mechanismů přijímajících (příkladem za všechny budiž překladače).

V tomto textu se nejprve seznámíme se základními pojmy teorie formálních jazyků, poté se zaměříme na obecný úvod do problematiky přijímajících gramatik a naznačíme, proč není příliš zajímavé zkoumat přijímající verze gramatik Chomského hierarchie. Následně se hlouběji ponoříme do zkoumání přijímajícího módu uspořádaných gramatik. Tyto gramatiky jsme vybrali jako názornou ukázku zajímavého zvýšení síly oproti obyčejnému, generujícímu módu.

Kapitola 2

Základní pojmy

Abeceda je konečná neprázdná množina symbolů. Řetězec nad abecedou Σ je posloupnost symbolů $w = a_1a_2 \dots a_n$, $\forall i = 1, \dots, n : a_i \in \Sigma$ a $|w| = n$ nazvěme délkou řetězce w. Existuje zvláštní řetězec ε takový, že $|\varepsilon| = 0$, nazýváme jej řetězcem prázdným. Množinu všech symbolů obsažených v řetězci w značíme $\mathrm{alph}(w)$. Σ^* značí množinu všech řetězců nad abecedou Σ. Jazyk L nad abecedou Σ je definován jako $L \subseteq \Sigma^*$. Jazyky L_1, L_2 budeme považovat za ekvivalentní právě tehdy, když $L_1 - \{\varepsilon\} = L_2 - \{\varepsilon\}$. Konkatenací jazyků L_1, L_2 rozumíme jazyk

$$L_1 \cdot L_2 = L_1L_2 = \{xy | x \in L_1, y \in L_2\}.$$

Dále nechť $\partial^l_w(L)$, respektive $\partial^r_w(L)$, označuje jazyk $\partial^l_w(L)\{w' | ww' \in L\}$, respektive $\partial^r_w(L)\{w' | w'w \in L\}$.

Gramatiky a jazyky Chomského hierarchie

(Generující) gramatika G je čtveřice $G = (V, T, P, S)$, kde

- V je množina symbolů (totální abeceda),
- T je množina terminálních symbolů (terminálů), $T \subset V$,

Typ	Název	Značení	Gramatiky	Omezení formy pravidel
0	rek. vyčíslitelné	$\mathcal{L}_0, \mathcal{L}(RE)$	neomezené	žádné další
1	kontextové	$\mathcal{L}_1, \mathcal{L}(CS)$	kontextové	$\lvert\alpha\rvert <= \lvert\beta\rvert$ (kromě $S \to \varepsilon$)
2	bezkontextové	$\mathcal{L}_2, \mathcal{L}(CF)$	bezkontextové	$\alpha \in N$
3	regulární	$\mathcal{L}_3, \mathcal{L}(REG)$	pravé lineární	$\alpha \in N, \beta \in T^* \cdot (N \cup \{\varepsilon\})$

Tabulka 2.1: Chomského hierarchie tříd jazyků; N zde označuje množinu neterminálů, tedy $N = (V - T)$.

- S je počáteční symbol, $S \in V - T$ a
- P je množina pravidel tvaru $\alpha \to \beta$, $\alpha, \beta \in V^*$, $\mathrm{alph}(\alpha) \cap (V - T) \neq \emptyset$.

Prvky množiny $(V - T)$ nazýváme neterminální symboly (neterminály).

Mějme gramatiku $G = (V, T, P, S)$. Říkáme, že řetězce $x = u\alpha v$ a $y = u\beta v$, $x, y \in V^*$, jsou v relaci přímé derivace, značíme $x \Rightarrow_G y$ nebo zkráceně $x \Rightarrow y$, právě tehdy, když $\alpha \to \beta \in P$. Pravidlo tvaru $\alpha \to \varepsilon$ se nazývá vymazávací nebo také ε-pravidlo. Nechť $\Rightarrow_G^*$ označuje reflexivní a tranzitivní uzávěr relace $\Rightarrow_G$. Jazyk generovaný gramatikou G, $L(G) = \{w \in T^* | S \Rightarrow_G^* w\}$.

Chomského hierarchie klade omezení na tvar pravidel a podle toho rozlišuje čtyři třídy jazyků (třídy jazyků značíme písmenem $\mathcal{L}$, aby nedošlo k záměně s jednotlivými jazyky) – viz tabulku 2.1. Platí, že $\mathcal{L}_3 \subset \mathcal{L}_2 \subset \mathcal{L}_1 \subset \mathcal{L}_0$.

Pokud máme třídu jazyků $\mathcal{L}(X)$, pak $\mathcal{L}(X - \varepsilon)$ označuje třídu jazyků definovaných mechanismem X bez použití vymazávacích pravidel, tedy například $\mathcal{L}(CF - \varepsilon)$ představuje třídu bezkontextových jazyků mimo jazyky využívající ε-pravidla.

Lemma 2.1. *(Kurodova normální forma) Pro každý kontextový jazyk L existuje gramatika $G = (V, T, P, S)$, $L(G) = L$, jejíž pravidla jsou v jednom z tvarů*

$$A \to BC, \qquad AB \to CD, \qquad A \to a.$$

kde $A, B, C, D \in (V - T)$, $a \in T$.

Kapitola 3

Přijímající gramatiky

V této kapitole se zaměříme na přijímající gramatiky a jejich vztah ke gramatikám generujícím.

Přijímající gramatika (budeme rovněž používat označení *přijímající verze* či *přijímající mód* gramatiky) se od generující gramatiky (*generující verze*, *generujícího módu*) liší ve dvou náležitostech:

1. Namísto počátečního symbolu/neterminálu je definován symbol cílový – pokud při derivacích dospějeme ke větné formě tvořené právě tímto jediným symbolem, derivace byla úspěšná a řetězec terminálů, z kterého jsme začali, *patří* do jazyka přijímaného danou gramatikou.

2. Uvažujeme-li nějakou třídu jazyků, obvykle je definována určitým omezením na tvar pravidel, která můžeme v gramatice použít, aby jí definovaný jazyk spadal do dané třídy. Tedy například bezkontextové gramatiky mohou obsahovat pouze pravidla tvaru $A \to w$, přičemž A je neterminální symbol a w je nějaký řetězec symbolů dané gramatiky. U přijímajících gramatik jsou tato omezení „otočena", tedy pro přijímající bezkontextové gramatiky dostáváme omezení na tvar pravidel $w \to A$, kde opět A je neterminální symbol a w je řetězec. Pokud (generující) bezkontextová gramatika obsahuje ε-pravidla, pak v přijímajícím módu lze použít poměrně nezvyklá pravidla tvaru $\varepsilon \to A$, tedy se nám mohou

na různých místech větné formy „zjevovat“ neterminální symboly.

Pomocí této charakteristiky můžeme ke generujícím gramatikám jednoduše vytvořit jejich přijímající protějšek, stačí interpretovat počáteční symbol jako symbol koncový a všechna pravidla tvaru $\alpha \to \beta$ nahradit pravidly $\beta \to \alpha$.

Relace přímé derivace nechť je v přijímajících gramatikách definována obdobně jako v gramatikách generujících, tedy řetězec $u\alpha v$ přímo derivuje řetězec $u\beta w$ (značíme rovněž $u\alpha v \Rightarrow_{G'} u\beta w$, kde G' je uvažovaná přijímající gramatika) právě tehdy, když v gramatice G' existuje pravidlo $\alpha \to \beta$ a zároveň jsou splněna další omezení vyžadovaná konkrétním typem řízeného přepisování.

Abychom odlišili příjímající mód od generujícího, budeme značit $L_g(G)$ jazyk generovaný gramatikou G a $L_a(G)$ jazyk přijímaný gramatikou G. Podobně pro třídy jazyků, $\mathcal{L}_g(X)$ bude označovat třídu jazyků generovaných a $\mathcal{L}_a(X)$ třídu jazyků přijímaných „mechanismem“ X (tedy například $\mathcal{L}_g(CF)$ představuje třídu jazyků generovaných bezkontextovými gramatikami a $\mathcal{L}_a(CF)$ představuje třídu jazyků přijímaných bezkontextovými gramatikami). Pokud nebude mód explicitně uveden, uvažujeme mód generující.

Obecné pozorování: Pokud $\mathcal{L}_g(X) \subseteq \mathcal{L}_g(Y)$, pak $\mathcal{L}_a(X) \subseteq \mathcal{L}_a(Y)$. Je tomu tak proto, že inkluzi $\mathcal{L}_g(X) \subseteq \mathcal{L}_g(Y)$ obecně dokazujeme tím způsobem, že každou derivaci $x \Rightarrow y$ mechanismu X nasimulujeme posloupností derivací $x \Rightarrow^* y$ mechanismu Y. Pak ale zjevně můžeme obdobnou simulací ukázat i $\mathcal{L}_a(X) \subseteq \mathcal{L}_a(Y)$.

Přijímající gramatiky a Chomského hierarchie

Věta 3.1. *$\mathcal{L}_g(X) = \mathcal{L}_a(X)$ pro $X \in \{REG, CF, CS, RE\}$.*

Důkaz. Důkaz je možno provést obecně pro takzvané *gramatiky s kontextovými podmínkami*. Gramatiky charakterizující jazyky Chomského hierarchie jsou pak jen speciálními případy těchto gramatik. Pro podrobný důkaz viz například [1]. □

Kapitola 4

Přijímající verze uspořádaných gramatik

V této kapitole si ukážeme, že uspořádané gramatiky v přijímajícím módu charakterizují větší třídu jazyků než uspořádané gramatiky v módu generujícím.

Definice 4.1. *Uspořádaná gramatika je pětice $G = (V, T, P, S, \prec)$, kde V je konečná množina symbolů, $T \subset V$ je množina terminálů, P je množina pravidel, S je axiom a $\prec$ je relace uspořádání na množině P.*

Mějme uspořádanou gramatiku $G = (V, T, P, S, \prec)$ a řetězec $x \in V^*$. Pravidlo $\alpha \to \beta \in P$ je aplikovatelné na řetězec x právě tehdy, když

- $x = w_1 \alpha w_2$ a zároveň
- x neobsahuje podřetězec α' takový, že $\alpha' \to \beta' \in P$ a $\alpha' \to \beta' \succ \alpha \to \beta$.

Relace $\prec$ nám tedy udává jakousi prioritu pravidel s tím, že nemůžeme aplikovat pravidlo, pokud existuje jiné pravidlo s vyšší prioritou, které bychom aplikovat mohli.

Jelikož definice uspořádané gramatiky sama o sobě neklade omezení na tvar pravidel, můžeme uvažovat pouze některé konkrétní podtřídy třídy jazyků generovaných uspořádanými gramatikami. Využijeme k tomuto účelu značení $\mathcal{L}(O, X)$, kde O říká, že jde o jazyky charakterizované uspořádanými gramatikami, a X označuje omezení typu pravidel například podle konkrétní podtřídy Chomského hierarchie.

Zamyslíme-li se nad silou přijímající verze těchto gramatik, musíme si nejdříve položit otázku, zda mají přijímající uspořádané gramatiky alespoň stejnou sílu, jako verze generující.

Musíme ukázat, že pro každou uspořádanou gramatiku v generujícím módu existuje přijímající verze charakterizující stejný jazyk.

Věta 4.2. *Ke každé uspořádané gramatice G s bezkontextovými pravidly v generativním módu existuje ekvivalentní uspořádané gramatika G s kontextovými pravidly v přijímajícím módu.*

Důkaz. Bez újmy na obecnosti předpokládejme, že gramatika

$$G = (V, T, P, S, \prec)$$

neobsahuje pravidla tvaru $A \to x$ a $A \to y$ taková, že $A \to x \prec A \to y$ (pravidlo s nižší prioritou, $A \to x$, by se nemohlo nikdy použít a generovaný jazyk by se jeho vypuštěním nezměnil). Dále pro každé pravidlo $A \to x \in P$, nechť $X_{A\to x}$ označuje množinu neterminálů A' takových, že $A \to x \prec A' \to y$ pro nějaké y. $X_{A\to x}$ tedy značí množinu neterminálů, které se ve větné formě nesmí vyskytovat, pokud chceme aplikovat pravidlo $A \to x$.

Ekvivalentní přijímající gramatiku zkonstruujeme jako

$$G' = (V', T, P', S, \prec')$$

takto:

- Nechť
$$V' = V \cup \{[w, A] | A \to w \in P\} \cup \{F\}$$
a nechť tyto množiny jsou po dvou disjunktní. Neterminál F je speciální „chybový“ symbol, který při svém výskytu ve větné formě zabrání vygenerování platné věty jazyka, neboť nebude existovat pravidlo, jež by jej přepsalo na jiný symbol, a tedy ve větné formě zůstane neterminál F navždy.

- Množinu P' a relaci $\prec'$ zkonstruujme tak, že pro každé pravidlo $A \rightarrow w \in P$ přidáme do množiny P' pravidla a do relace $\prec'$ vztahy

    1. $w \rightarrow [w, A] \prec' [w, A] \rightarrow F$,
    2. $[w, A] \rightarrow A$ tak, že $\forall A' \in X_{A \rightarrow w} [w, A] \rightarrow A \prec' A' \rightarrow F$ a
    3. $\forall u \rightarrow V \in P', V \neq F, u \neq [w, A]$ nechť $u \rightarrow V \prec' [w, A] \rightarrow F$.

Neformálně bychom mohli říci, že gramatika G' rozděluje přepis pravidla v gramatice G na dvě fáze, přičemž díky tomuto mezikroku a struktuře relace ověří, zda se ve větné formě nevyskytuje neterminál, který má být přepsán až později. Takto může provádět simulaci gramatiky G v opačném pořadí derivací.

Konkrétně neterminály ve tvaru $[w, A]$ reprezentují specifické neterminály pro označení mezikroku, který právě probíhá – pokud se ve větné formě nachází neterminál $[w, A]$, znamená to, že právě probíhá simulace přepisu pravidla $A \rightarrow w$. Relace potom zajišťují, abychom nezačali stejné pravidlo simulovat vícekrát zároveň (bod 1), abychom různá pravidla nesimulovali zároveň bez dokončení mezikroku (bod 3) a aby derivace proběhly podle opačných priorit než ve verzi generující (bod 2). Ukažme si to na jednoduchém příkladu.

Příklad 4.1. *Nechť v generující gramatice existují pravidla $A \rightarrow x \succ B \rightarrow y$ a žádná jiná pravidla s A či B na levé straně. Pak máme zajištěno, že derivace větné formy AB bude probíhat v pořadí*

$$AB \Rightarrow xB \Rightarrow xy.$$

V tomto případě nebude možné simulovat pravidla ve stejném pořadí v gramatice G', neboť při derivaci

$$xy \Rightarrow [x, A]y \Rightarrow Ay \Rightarrow A[y, B]$$

musíme v následném kroku přepsat $[y, B]$ na chybový symbol F, protože $A \in X_{B \rightarrow y}$ a tedy $A \rightarrow F \succ' [y, B] \rightarrow B$.

Dokažme nejprve, že $L_g(G) \subseteq L_a(G')$. Stačí nám ukázat, že pokud $x \Rightarrow_G y$, pak $y \Rightarrow_{G'}^* x$. Musí tedy existovat pravidlo $A \to w \in P$ a řetězce x, y musí být ve tvaru $x = uAv$ a $y = uwv$. Zároveň musí platit, že se v řetězci x nevyskytuje žádný neterminál A' takový, že $A' \in X_{A \to w}$. V takovém případě můžeme provést derivaci $y \Rightarrow_{G'}^* x$ ve dvou krocích. Nejprve přepíšeme řetězec uwv na $u[w, A]v = z$ a poté $z = u[w, A]v$ na $uAv = x$ s tím, že díky vhodné konstrukci relace $\prec'$ máme zajištěno, že žádný neterminál z $X_{A \to w}$ se nenachází v řetězci z a tedy nemohlo dojít k porušení priorit pravidel.

Dále dokážeme, že $L_g(G) \supseteq L_a(G')$. Ukážeme, že pokud $y \Rightarrow_{G'} z \Rightarrow_{G'} x \Rightarrow_{G'}^* S$ pro $x, y \in V, z \in V'$, pak $x \Rightarrow_G y$. Řetězec y musí být ve tvaru $y = uwv$ a řetězec x musí být ve tvaru $x = uAv$. Pak je řetězec z nutně ve tvaru $z = u[w, A]v$ pro nějaké $A \to w \in P$ (po prvním kroku derivace by sice mohlo nastat, že $z = uFv$, nicméně v takovém případě se nedobereme platné věty jazyka a nedojdeme k axiomu S). Zároveň je zajištěno, že žádný neterminál z $X_{A \to w}$ se nevyskytuje v řetězci z, neboť pak by větší prioritu mělo pravidlo přepisující daný neterminál na neterminál chybový. Vidíme tedy, že derivace $y \Rightarrow_{G'} z \Rightarrow_{G'} x$ v situaci, kdy můžeme dojít k axiomu, nutně simuluje aplikaci pravidla $A \to w$ v gramatice G. $\square$

Ukažme si jako cvičení příklad, kde tato konstrukce „selže“. Odpověď na otázku, proč tomu tak je, ponecháváme na zvídavém čtenáři.

Příklad 4.2. *Mějme uspořádanou gramatiku $G = (\{A\}, \{a\}, P, A, \prec)$, kde množina P obsahuje dvě pravidla v relaci $A \to aa \succ A \to a$. Zřejmě tedy $L_g(G) = \{aa\}$. Přijímající gramatika G' obsahuje následující pravidla* *v relacích:*

$$
\begin{aligned}
aa &\to [aa, A] \prec' [aa, A] \to F \\
[aa, A] &\to A \\
a &\to [a, A] \prec' [a, A] \to F \\
[a, A] &\to A \prec' A \to F \\
aa &\to [aa, A] \prec' [a, A] \to F \\
a &\to [a, A] \prec' [aa, A] \to F
\end{aligned}
$$

Dostáváme tedy tyto možné úspěšné derivace v gramatice G':

$$
\begin{aligned}
aa &\Rightarrow [aa, A] \Rightarrow A \\
a &\Rightarrow [a, A] \Rightarrow A
\end{aligned}
$$

Dostáváme tedy $L_a(G') = \{a, aa\}$, tedy $L_g(G) \neq L_a(G')$! Proč?

Dokázali jsme tedy, že přijímající verze uspořádaných gramatik s bezkontextovými pravidly je minimálně stejně silná, jako verze generující. Nyní si ukážeme, že přijímající verze je dokonce striktně silnější. Intuitivně bychom důvod tohoto faktu mohli popsat takto: *Zatímco generující verze uspořádaných gramatik nám umožňuje kontrolovat pouze výskyt jednotlivých neterminálů, verze přijímající dokáže testovat výskyt celých řetězců.* Tato vlastnost se hodí například pro testování kontextu podřetězců, které se chystáme derivovat, s použitím pomocných pravidel existujících právě jen pro toto testování. TODO dopsat strana 13

Lemma 4.3. *Třídy jazyků $\mathcal{L}(O, CF)$ a $\mathcal{L}(O, CF-\varepsilon)$ jsou uzavřeny vzhledem ke sjednocení.*

Důkaz. Mějme dvě uspořádané gramatiky $G_1 = (V_1, T_1, P_1, S_1, \prec_1)$ a $G_2 = (V_2, T_2, P_2, S_2, \prec_2)$. Sestrojme gramatiku G, $L_a(G) = L_a(G_1) \cup L_a(G_2)$. Bez újmy na obecnosti předpokládejme, že množiny neterminálů gramatik G_1, G_2 jsou disjunktní ($(V_1 - T_1) \cap (V_2 - T_2) = \emptyset$) a neobsahují neterminál S ($S \notin (V_1 - T_1)$, $S \notin (V_2 - T_2)$). Gramatiku G pak zkonstruujeme jako

$$G = (V_1 \cup V_2 \cup \{S\}, T_1 \cup T_2, P_1 \cup P_2 \cup \{S_1 \to S, S_2 \to S\}, \prec_1 \cup \prec_2).$$

$\square$

Lemma 4.4. $\mathcal{L}(CF-\varepsilon) \subset \mathcal{L}(O, CF-\varepsilon)$.

Důkaz. Důkaz, že $\mathcal{L}(CF-\varepsilon) \subseteq \mathcal{L}(O, CF-\varepsilon)$, je zřejmý – stačí použít prázdnou relaci uspořádání a pro každou bezkontextovou gramatiku dostaneme ekvivalentní uspořádanou gramatiku s bezkontextovými pravidly. Dále existuje jazyk $L = \{a^{2^n} | n = 0, 1, \dots\}$, který nepatří do $\mathcal{L}(CF-\varepsilon)$, ale patří do $\mathcal{L}(O, CF-\varepsilon)$ [2]. Generuje jej totiž například uspořádaná gramatika $G = (\{A, B, C, D, E, F, a\}, \{a\}, P, A, \prec)$, kde množina P a relace $\prec$ jsou definovány takto:

$$
\begin{aligned}
A \to C \prec B \to F \prec C \to F \prec D \to F \prec E \to F \\
C \to B \prec A \to B \prec D \to F \prec E \to F \\
B \to D \prec A \to F \prec C \to F \prec D \to F \prec E \to F \\
D \to AA \prec B \to AA \prec C \to F \prec E \to F \\
A \to E \prec B \to F \prec C \to F \prec D \to F \prec E \to F \\
E \to a \prec A \to a \prec B \to F \prec C \to F \prec D \to F
\end{aligned}
$$

$\square$

Lemma 4.5. $\mathcal{L}_g(CF) \subset \mathcal{L}_a(O, CF-\varepsilon)$.

Důkaz. Protože $\mathcal{L}_g(CF) = \mathcal{L}_g(CF-\varepsilon)$ a zároveň dle věty 4.2 $\mathcal{L}_g(O, CF-\varepsilon) \subseteq \mathcal{L}_a(O, CF-\varepsilon)$, dostáváme s použitím lemmatu 4.4 inkluzi $\mathcal{L}_g(CF) \subset \mathcal{L}_a(O, CF-\varepsilon)$. $\square$

Věta 4.6. $\mathcal{L}_a(O, CF-\varepsilon) = \mathcal{L}_g(CS)$.

Důkaz. Inkluzi $\subseteq$ dokážeme jednoduše konstrukcí lineárně omezeného automatu (LOA, viz [5]) přijímajícího stejný jazyk. Mějme uspořádanou gramatiku G s bezkontextovými pravidly bez ε-pravidel. Automat bude pracovat takto:

1. Vybere pravidlo $w \to A$, jehož aplikace se bude simulovat.

2. Ověří, zda posloupnost symbolů na pásce obsahuje podřetězec w (páska se nachází ve tvaru $\Delta x \Delta \dots \Delta$, kde x je aktuální větná forma).

3. Pro všechny $w', A' : w' \to A' \succ w \to A$ ověří, zda páska neobsahuje podřetězec w'.

4. V případě, že oba testy uspějí, LOA nedeterministicky vybere jeden z výskytů řetězce w a nahradí jej řetězcem $A\Delta_{|w|-1}$. Pokud alespoň jeden z testů selže, přejde do bodu 1.

5. Automat přesune nově vytvořené symboly Δ napravo od větné formy, takže se páska dostane tvaru $\Delta y \Delta \dots \Delta$, kde y je nová větná forma. Tím jsme tedy nasimulovali derivaci $x \Rightarrow_G y$.

6. Ověří, zda se páska nenachází ve tvaru $\Delta S \Delta \dots \Delta$ – pokud ano, přejde do koncového stavu, jinak pokračujeme bodem 1.

Důkaz inkluze $\supseteq$ je poněkud komplikovanější. Mějme jazyk typu 1 $L \in \mathcal{L}(CS)$, $L \subseteq T^+$. Tento jazyk si můžeme definovat jako sjednocení jazyků

$$L = \bigcup_{a,b \in T} \{a\}\partial_a^l(\partial_b^r(L))\{b\} \cup (L \cap (T \cup T^2)),$$

díky čemuž nám postačí dokázat, že $\{a\}M\{b\} \in \mathcal{L}_a(O, CF - \varepsilon)$ pro $M \in \mathcal{L}_g(CS)$, $\varepsilon \notin M$ [3] (prázdný řetězec nemusíme v jazyce M uvažovat, jelikož jazyk $\{ab\}$ je součástí druhé části sjednocení). Nechť $G = (V, T, P, S)$ je kontextová gramatika v Kurodově normální formě, která generuje jazyk M, tedy $L_g(G) = M$. Označme každé pravidlo tvaru $XU \to YZ$ unikátním návěštím r (značeno $r : XU \to YZ$) a označme množinu těchto návěští Lab. Nyní můžeme zkonstruovat uspořádanou gramatiku s bezkontextovými pravidly bez epsilon pravidel přijímající jazyk $\{a\}M\{b\}$,

$$G = (V', T', P', S', \prec),$$

$L_a(G) = \{a\}M\{b\}$:
$$T' = T \cup \{a, b\}$$
$V' = (V - T) \cup T' \cup \{A, B, S', F\} \cup \{(A, r), [A, r], (Y, r), [Z, r] | r : XU \to YZ \in P\}$

Bez újmy na obecnosti předpokládejme, že jednotlivé množiny tvořící množinu V' jsou disjunktní. Množinu pravidel P' tvoří tato pravidla:

1. Pravidla použitá pro započetí a ukončení derivace a pravidla simulující aplikaci bezkontextových pravidel:

   (a) $a \to A \prec \{y \to F | y \in (V' - T')\}$

   (b) $b \to B \prec \{y \to F | y \in (V' - (T' \cup \{A\}))\}$

   (c) $x \to X \prec \{(A, r) \to F, [A, r] \to F | r \in Lab\}$ pro všechna bezkontextová pravidla $X \to x \in P$

   (d) $ASB \to S'$

2. Dále pro všechna kontextová pravidla tvaru $r : XU \to YZ \in P$ množina P' obsahuje následující pravidla, jež simulují aplikace pravidla r v gramatice G:

   (a) $A \to [A, r] \prec \{(A, s) \to F, [A, s] \to F | s \in Lab\}$

   (b) $Y \to [Y, r] \prec \{A \to F, (A, s) \to F, [A, s'] \to F, [R, s] \to F, (R, s) \to F | s \in Lab, s' \in (Lab - \{r\}), R \in (V - T)\}$

   (c) $Z \to (Z, r) \prec \{A \to F, (A, s) \to F, [A, s'] \to F, [R, s'] \to F, (R, s) \to F | s \in Lab, s' \in (Lab - \{r\}), R \in (V - T)\}$

   (d) $[A, r] \to (A, r) \prec \{A \to F, (A, s) \to F, [A, s'] \to F, [R, s'] \to F, (R, s') \to F, z(Z, r) \to F, [Y, r]y \to F | s \in Lab, s' \in (Lab - \{r\}), R \in (V - T), z \in (V' - \{[Y, r]\}), y \in (V' - \{(Z, r)\}\}$ (Zde můžeme vidět, že ověřujeme levý a pravý kontext určitých neterminálů, kvůli čemuž potřebujeme mít označen začátek a konec větné formy pomocnými neterminály A, B.)

(e) $(A, r) \to A \prec (Z, r) \to U \prec [Y, r] \to X \prec \{A \to F, [A, s] \to F, (A, s') \to F, [R, s'] \to F, (R, s') \to F | s \in Lab, s' \in (Lab - \{r\}), R \in (V - T)\}$

Obecně úspěšná derivace probíhá takto: Nejprve se přepíší symboly na začátku a konci řetězce (a a b) na neterminály (A a B). Přítomnost neterminálu A ve větné formě indikuje, že je možné započít simulaci kteréhokoliv pravidla z G nebo aplikovat koncové pravidlo $ASB \to S'$. Jelikož simulace aplikace bezkontextových pravidel je velmi přímočará (aplikuje se pravidlo popsané v bodu 1c), detailně si popíšeme simulaci aplikace pravidel kontextových:

1. Nejdříve použijeme pravidlo $A \to [A, r]$, čímž určíme, že budeme simulovat právě pravidlo s návěštím r.

2. Nalezneme jeden výskyt neterminálu Y, respektive Z a přepíšeme jej na neterminál $[Y, r]$, respektive (Z, r). Vhodně zkonstruovaná relace $\prec$ nám zábrání přepsat některý z neterminálů vicekrát než jednou.

3. Fáze výběru přepisovaných neterminálů je ukončena aplikací pravidla $[A, r] \to (A, r)$. Díky vhodnému nastavení priorit je zajištěno, že se vybrané neterminály Y a Z nacházejí vedle sebe. Pravidlo $[Y, r]y \to F$ ověřuje, zda symbol vpravo od neterminálu $[Y, r]$ není neterminál různý od $z(Z, r)$. Zároveň se nemůže stát, aby byl neterminál na konci větné formy, a to díky pomocnému „hraničnímu“ neterminálu B[1] Podobně pravidlo $z(Z, r) \to F$ ověří, zda není symbol nalevo od (Z, r) různý od $[Y, r]$. Poznamenejme, že je potřeba uskutečnit obě tyto kontroly, abychom zabránili „zkratkám“ spočívajícím v nahrazení pouze jednoho z neterminálů Y, Z. Povšimněme si, že toto je jediná situace, kdy potřebujeme ověřit (ne)přítomnost řetězce a ne jen neterminálu.

[1]Použití neterminálu B bychom se mohli vyhnout, kdybychom celý důkaz postavili na převodu gramatiky v Penttonenově normální formě (viz například [4]) namísto Kurodovy.

4. Nakonec použijeme pravilda z části 2e pro dokončení simulace přepisu kontextového pravidla. Tím se nám mimo jiné stane z neterminálu (A, r) neterminál A, jenž nám umožní započít simulaci aplikace dalšího pravidla.

□

Pokud si v tuto chvíli uvědomíme, uspořádané gramatiky bez ε-pravidel jsou stejně silné jako gramatiky s náhodným kontextem bez ε-pravidel, které jsou ostře slabší než kontextové gramatiky [3], docházíme k následujícímu závěru:

Důsledek 4.7. $\mathcal{L}_g(CF) \subset \mathcal{L}_g(O, CF - \varepsilon) \subset \mathcal{L}_g(O, CF - \varepsilon)$

Věta 4.8. $\mathcal{L}_g(O, CF) \subseteq \mathcal{L}_a(O, CF) = \mathcal{L}_g(RE)$

Důkaz. Postačí nám ukázat, že $\mathcal{L}_g(RE) \subseteq \mathcal{L}_a(O, CF)$. Podle [5] každý rekurzivně vyčíslitelný jazyk $L \subseteq \Sigma^*$ je homomorfním obrazem nějakého kontextového jazyka $L' \subseteq \Gamma$, tedy $L = h(L')$ při morfismu $h : \Gamma^* \to \Sigma^*$. Bez újmy na obecnosti předpokládejme, že $\Sigma \cup \Gamma = \emptyset$. Dále nechť $\Gamma' = \{a' | a \in \Gamma\}$. Inkluzi pak můžeme dokázat podobným způsobem jako u věty 4.6 s tím, že navíc přidáme pravidla

- $w \to a' \prec \{b \to F | b \in \Gamma\}$ pro všechna $w = h(a)$,
- $a' \to a$ pro všechna $a \in \Gamma$ a
- $c \to F$ pro všechna $c \in \Sigma \cup \Gamma'$ tak, aby tato pravidla měla vždy vyšší prioritu než pravidla v důkazu věty 4.6.

Nejprve tedy při přijímání řetězce výslednou gramatikou postupně „inverzně aplikujeme“ morfismus h a poté postupujeme stejně jako v gramatice vytvořené ve větě 4.6. □

Nyní si opět uvědomme, že uspořádané gramatiky s bezkontextovými pravidly jsou stejně silné jako gramatiky se zakazujícím kontextem [3], které jsou ostře slabší než jazyky rekurzivně vyčíslitelné. Naproti tomu jsme ukázali, že přijímající mód uspořádaných gramatik s bezkontextovými pravidly má sílu stejnou jako rekurzivně vyčíslitelné jazyky. Dostáváme tedy následující závěr:

Důsledek 4.9. $\mathcal{L}_g(O, CF) \subset \mathcal{L}_a(O, CF)$.

Kapitola 5

Závěr

V tomto textu jsme se zaměřili především na důkladný rozbor přijímajícího módu uspořádaných gramatik. Ukázali jsme, že jednoduchou změnou pohledu na notoricky známé mechanismy generování jazyků se dostáváme do zajímavé oblasti, ve které se mohou skrývat poměrně neočekávané výsledky, týkající se především zvýšení síly zkoumaného mechanismu.

Ačkoliv jsme se drželi formy matematického textu, pokusili jsme se při tom nastínit i určité intuitivní představy, jež mohou mnohem lépe postihnout a objasnit jádro problému, především laickému čtenáři.

Celkově jsou přijímající gramatiky oblastí velmi zajímavou a popsání veškerých detailů by vyžadovalo mnohem větší prostor, než je rozsah této seminární práce. Zájemcům o další studium vřele doporučujeme text [1], který se snaží pokrýt problematiku s mnohem větším záběrem a obsahuje i mnoho užitečných referencí pro další studium.

Literatura

[1] Bordihn, H., Fernau, H.: Accepting Grammars and Systems (1995)

[2] Stockij, È.D.: Upravlenie vyvodom v formal'nych grammatikah. Problemy peredači informacii **7**(3) (1971) 87–102

[3] Dassow, J., Păun, G.: Regulated Rewriting in Formal Language Theory EATCS Monographs in Theoretical Computer Science 18 (1989)

[4] Penttonen, M.: One-sided and two-sided context in formal grammars Information and Control 25 (1974) 371–392 Regulated Rewriting in Fromal Language Theory EATCS Monographs in Theoretical Computer Science 18 (1989)

[5] Salomaa, A.K.: Formal Languages Academic press (1973)