



KATEDRA INFORMATIKY
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
UNIVERZITA KOMENSKÉHO, BRATISLAVA

VLASTNOSTI BEZKONTEXTOVÝCH GRAFOVÝCH JAZYKOV

(Diplomová práca)

MICHAL BORGA

Vedúci: prof. RNDr. Branislav Rován, PhD.

Bratislava, 2008

**Vlastnosti bezkontextových
grafových jazykov**

DIPLOMOVÁ PRÁCA

Michal Borga

**UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
KATEDRA INFORMATIKY**

Informatika

Vedúci diplomovej práce
prof. RNDr. Branislav Rován, PhD.

BRATISLAVA 2008

Čestne prehlasujem, že som túto diplomovú prácu
vypracoval samostatne s použitím citovaných zdro-
jov.

.....

Podakovanie

Ďakujem môjmu vedúcemu diplomovej práce prof. RNDr. Branislavovi Rovanovi, PhD. za poskytnutú podporu, trpezlivosť a jeho cenné rady pri písaní tejto práce. Ďalej sa chcem poďakovať mojej rodine a priateľom, ktorí ma podporovali počas písania tejto práce a počas celého môjho štúdia.

Bibliografická identifikácia

Autor: Michal Borga
Názov diplomovej práce: Vlastnosti bezkontextových grafových jazykov
Škola: Univerzita Komenského v Bratislave
Fakulta: Fakulta matematiky, fyziky a informatiky
Katedra: Katedra informatiky
Vedúci diplomovej práce: prof. RNDr. Branislav Rován, PhD.
Rozsah práce: 65 strán

Bratislava, máj 2008

Abstrakt

V práci sa zaoberáme skúmaním vlastností a porovnávaním rôznych typov NCE a NLC grafových jazykov. Predstavíme si analogické aj kontrastné vlastnosti niektorých tried grafových jazykov v porovnaní s jazykmi nad reťazcami. Napríklad si ukážeme, že bezkontextové grafové jazyky spĺňajú vlastnosti pumpovacej lemy pre bezkontextové jazyky nad reťazcami, čo sa týka ich hustoty slov v jazyku. A naopak ukážeme, že bezkontextové grafové jazyky nemôžu obsahovať všetky grafy nad danými abecedami uzlov a hrán.

Kľúčové slová

grafové gramatiky, miery zložitosti, blokujúce hrany, konfluentnosť, usmerňovanie odvodenia

Predhovor

História prepisovacích systémov grafov začala pred takmer štyridsiatimi rokmi. Od začiatku boli systémy na prepisovanie grafov používané a vyvíjané pre riešenie reálnych problémov na poli informatiky, biológie a podobne. Grafové gramatiky sú zaujímavé aj z teoretického hľadiska, pretože sú prirodzeným zovšeobecnením teórie formálnych jazykov nad reťazcami a teórie prepisovania termov založenej na stromoch. Široké využitie grafových gramatík je založené na fakte, že grafy sú prirodzeným spôsobom popisovania komplexných situácií na intuitívnej úrovni. Navyše, transformácie grafov spojené s prepisovaním grafovou gramatikou vnášajú dynamiku do takýchto popisov, pretože modelujú vývoj grafových štruktúr. Preto sa stali grafové gramatiky a grafové transformácie atraktívne napríklad ako programovacia paradigma pre softvérové a grafické rozhrania. Ďalšou veľmi zaujímavou oblasťou ich využitia je hĺbková analýza dát (data mining). [2, 5]

Oblasti praktického využitia grafových transformácií:

- transformácie modelov a programov
- syntax a sémantika vizuálnych jazykov
- modelovanie, meta-modelovanie a MDA (model-driven architecture)
- softvérová architektúra a vývoj
- „refactoring“ programov a softvérových systémov
- bezpečnostná politika

- XML

Z teoretického hľadiska sa môžeme na grafový jazyk, podobne ako v teórii formálnych jazykov, pozerat' ako na problém a na jeho slová ako inštalácie problému. Aj preto je dôležité túto abstrakciu formálnych jazykov pochopiť a poznať jej komplikovanosť, mnohotvárnosť a spleť. Inak povedané, je dôležité skúmať jej miery zložitosti. Celá táto snaha sa dá v podstate chápať ako pokus o nájdenie hierarchie a usporiadania jednotlivých tried grafových jazykov na základe ich jedinečných vlastností. Cieľom tejto práce je prezentovať základné typy bezkontextových grafových gramatík a preskúmať vlastnosti nimi generovaných jazykov, z hľadiska zložitosti usporiadaných postupností grafov v týchto jazykoch.

Obsah

1	Úvod	1
2	Grafové gramatiky a ich vlastnosti	3
2.1	NLC (Node Label Controlled) grafové gramatiky	4
2.2	NCE grafové gramatiky	6
2.3	Ďalšie triedy grafových gramatík	11
2.4	Normálne tvary C-edNCE gramatík	12
2.5	Charakterizácia C-edNCE triedy jazykov	16
2.5.1	Logická charakterizácia	16
2.5.2	Príklady využitia MSO charakterizácie	18
2.6	Uzáverové vlastnosti bezkontextových grafových jazykov . . .	21
3	Regulárne grafové gramatiky...	22
4	Kvantitatívne vlastnosti grafov...	27
4.1	Vlastnosti grafov generovaných grafovými gramatikami	28
4.2	Zhrnutie	38
4.2.1	Zhrnutie vlastností	39
5	Usmerňovanie prepisovania	42
5.1	Blokujúce hrany a regulárne gramatiky riadenia odvodenia v C-edNCE gramatikách	42
5.1.1	Skúmané vlastnosti	44

5.1.2	Porovnanie vlastností regulárnych gramatík riadenia odvedenia a blokujúcich hrán nad C-edNCE prepisovacím systémom:	48
5.1.3	Zhrnutie	52
5.2	Blokujúce hrany a regulárne gramatiky riadenia odvedenia v edNCE gramatikách	52
5.2.1	Porovnanie vlastností regulárnych gramatík riadenia odvedenia a blokujúcich hrán nad edNCE prepisovacím systémom	52
5.3	Zhrnutie	54
5.3.1	Otvorené otázky k výsledkom kapitoly	55
6	Hierarchie edNCE jazykov	56
6.1	Vlastnosti hierarchie edNCE	57
6.1.1	Otvorené otázky:	58
6.2	Vlastnosti hierarchie NLC	58
6.2.1	Zhrnutie:	59
7	Záver	60

Zoznam obrázkov

2.1	(a) pravidlá gramatiky, (b) odvodenia gramatiky	7
2.2	Konfluencia resp. Church Roserova vlasnosť	9
2.3	Konfluencia	10
2.4	Hierarchia edNCE jazykov.	13
3.1	Ilustračný obrázok starého pravidla a nových pravidiel. (a) staré pravidlo, (b) nové pravidlá.	25
3.2	Ilustračný obrázok prepisu starého pravidla novými pravidla- mi. (a) staré pravidlo, (b) prepis novými pravidlami.	26
4.1	(a) p-strom odvodenia, (b) c-strom odvodenia	29
4.2	Pravidlá pre gramatiku	37
5.1	Vetné formy jazyka L	46
5.2	Gramatika pre jazyk L	47
6.1	Otázka vlastností NLC jazykov, ktoré nie sú BNLC	58

Kapitola 1

Úvod

Jazyk v teórii formálnych jazykov predstavuje množinu slov. *Grafový jazyk* je také zovšeobecnenie jazyka pre reťazce, ktorého slová tvoria grafy. Dva základné spôsoby skúmania grafových jazykov sú *generatívny*, pomocou gramatík a *rozpoznávací*, pomocou automatov. V jazykoch Chomského hierarchie nad reťazcami generovanie a rozpoznávanie predstavujú duálne úlohy. Pre grafové jazyky dualita medzi generovacími a rozpoznávacími zariadeniami, ako je to pre jazyky v Chomského hierarchii, vo všeobecnosti nie je známa (okrem LIN-NCE a B-NCE). Aj táto vlastnosť grafových jazykov naznačuje väčšie problémy pri ich skúmaní, spojené so zložitou štruktúrou grafov v porovnaní s jazykmi nad reťazcami. Nasledujúce kapitoly tejto práce budú venované generatívnemu prístupu skúmania grafových jazykov.

Grafové transformácie sa vo všeobecnosti (podľa prístupu skúmania) delia na (kvôli jednoznačnosti uvádzame aj v anglickej terminológii):

- Nahrádzanie uzlov (Node label replacement)
- Nahrádzanie hrán (Hyperedge replacement)
- Algebraický prístup (Algebraic)
 - Double Pushout (DPO)
 - Single Pushout (SPO)
 - High Level Replacement (HLR)

- Logický prístup (Logic)
- Teória 2 štruktúr (Theory of 2-structures)
- Programované prepisovanie grafov (Programmed graph replacement)

V tejto práci budú grafové transformácie spomínané najmä z pohľadu nahrádzania uzlov (Node Label Replacement).

Kapitola 2

Grafové gramatiky a ich vlastnosti

V tejto kapitole uvádzame základné definície a známe výsledky z oblasti grafových gramatík. Našou snahou je uviesť do tejto problematiky čitateľa, ktorý pozná klasickú teóriu formálnych jazykov zaoberajúcu sa slovami. Grafové gramatiky tvoria zovšeobecnenie gramatík z gramatík pre reťazce na gramatiky pre generovanie grafov. Grafové gramatiky patria ku generatívnym prístupom skúmania grafových jazykov. Generovanie grafov pomocou gramatík je založené na opakovanom prepisovaní grafov podľa pravidiel gramatiky. My budeme uvažovať prepis grafu vo forme nahradenia jeho uzla. Prepis uzla podľa pravidla gramatiky v znamená vznik novej časti grafu. Túto treba akýmsi spôsobom začleniť do vetnej formy (grafu), ktorá bola doteraz vygenerovaná gramatikou. Z hľadiska začlenenia tejto časti pri prepisovaní grafov rozlišujeme *nahrádzanie uzlov*, ktoré pri prepisovaní používa *spájanie* (connecting) so zvyškom grafu a *nahrádzanie hrán*, ktoré využíva *lepenie* (gluing) stotožňovanie so zvyškom grafu. Podobne ako je to u jazykov pre reťazce aj tu prebehla snaha o hierarchizáciu jazykov nad rôznymi prepisovacími systémami. Tieto snahy však neviedli ku žiadnym významnejším výsledkom v porovnaní s jazykmi nad reťazcami, kde výsledkom týchto snáh bola Chomského hierarchia. Pri skúmaní rôznych typov grafových gramatík sa ako najvhodnejšie ukázali bezkontextové grafové gramatiky. Presnejšie HR gramatiky (hyperedge replacement grammars) pre prepisovanie hrán a C-edNCE gramatiky a ich podtriedy pre prepisovanie uzlov. Pravidlá pre

tieto gramatiky majú vo všeobecnosti tvar trojice (M, D, E)

- M - materský graf
- D - dcérsky graf
- E - mechanizmus vnorenia

V pravidlách *bezkontextových grafových gramatík* množina M pozostáva z jedného uzla alebo jednej hrany (nejakej atomickej časti). Týmto je zaručená lokálnosť a atomickosť prepisu pomocou gramatiky. Ďalšími vlastnosťami bezkontextových gramatík sú: prepis pri odvodzovaní je bez podmienok na použitie pravidiel a výsledok nahrádzania je nezávislý na poradí použitých pravidiel. Posledná vlastnosť bola pre bezkontextové jazyky nad reťazcami zaručená tým, že sa pri nich nemusel brať ohľad na priestorovú štruktúru slova. Na rozdiel od grafov, kde „slová“ nemusia mať rovnakú priestorovú štruktúru, všetky slová z jazyka nad reťazcami ju majú rovnakú a to v poňatí grafov reťaz. Uvedené vlastnosti bezkontextových grafových gramatík ich robia vhodnými pre popis rekurzívne definovaných množín grafov, resp. rekurzívnych vlastností grafov. Ich nevýhodou je viac zovšeobecňujúcich spôsobov definovania. Napríklad v spájacom mechanizme nahrádzania uzlov existuje niekoľko prirodzených spôsobov definovania vnáracieho mechanizmu E a niekoľko prirodzených obmedzení na pravé strany pravidiel D . V tejto kapitole si predstavíme mechanizmy prepisovania NLC a NCE spolu s ich bezkontextovými vlastnosťami.

2.1 NLC (Node Label Controlled) grafové gramatiky

NLC predstavuje jeden z najjednoduchších mechanizmov prepisovania grafov. Pravidla NLC gramatiky sú tvaru $X \rightarrow D$, čo znamená, že uzol so značkou X môže byť podľa pravidla nahradený dcérskym grafom D . Pravidlo sa môže aplikovať na akýkoľvek uzol so značkou X (neexistujú žiadne podmienky na použitie pravidla). Všetky pravidlá zdieľajú tú istú sadu spájacích inštrukcií C , ktorá je vo všeobecnosti definovaná ako binárna relácia

nad značkami uzlov v gramatike. Spájanie v NLC je lokálne. To znamená, že uzly dcérskeho grafu D sa môžu spojiť len takými uzlami materského grafu M , ktoré boli spojené hranou s prepisovaným uzlom X (uzly susedné s X vo vetnej forme pred použitím pravidla). NLC grafové gramatiky sú bezkontextové v zmysle, že ich prepisovanie je lokálne a nemajú žiadne podmienky na použitie pravidiel. Tieto vlastnosti umožňujú ich odvodenia modelovať na stromoch odvodenia. Ktoré stelesňujú rekurzívny charakter bezkontextových gramatík. I keď vo všeobecnosti NLC gramatiky nemajú žiaducu bezkontextovú vlastnosť. Ich strom odvodenia totiž nie je nezávislý na poradí použitia pravidiel. Grafová gramatika, ktorá má túto vlastnosť sa nazýva konfluentná (confluent), alebo majúca Church-Rosserovu vlastnosť.

Nasledujúci príklad ilustruje možnosť vzniku rôznych grafov použitím rovnakých pravidiel v odvodení pri absencii konfluentnosti v NLC gramatikách. Uvedieme ho pre ilustráciu tejto myšlienky ešte pred formálnou definíciou NLC gramatík. Spoliehame sa na analógiu s gramatikami nad reťazcami.

Príklad 2.1.1. Majme gramatiku G s počiatočným grafom S a pravidlami $S \rightarrow AB$, $A \rightarrow a$, $B \rightarrow b$ (S , A , B sú neterminály a a , b sú terminály). Keď sa na túto gramatiku pozrieme ako na bezkontextovú gramatiku pre reťazce, $L(G) = \{ab\}$ a G má práve jeden strom odvodenia. Predpokladajme teraz, že G má reláciu spojení $C = \{(B, a), (a, b)\}$. Potom $L(G) = \{ab, a + b\}$, kde $a + b$ je graf s dvoma izolovanými uzlami a a b . G má stále jeden strom odvodenia, ale dve rôzne inštancie tohto stromu, ktoré vzniknú jeho rôznym prechádzaním. Odvodenia:

1. $S \rightarrow AB$, $A \rightarrow a$, $B \rightarrow b$.
2. $S \rightarrow AB$, $B \rightarrow b$, $A \rightarrow a$.

Pre postupnosť pravidiel v 1. vznikne ab a v 2. vznikne $a + b$. Preto G nie je konfluentná (confluent).

Presná definícia NLC grafovej gramatiky znie nasledovne:

Definícia 2.1.1. NLC grafová gramatika je 5-tica $G = (\Sigma, \Delta, P, C, S)$ kde $\Sigma - \Delta$ je abeceda neterminálnych značiek uzlov a Δ ($\Delta \subseteq \Sigma$) je abeceda terminálnych značiek uzlov. P je konečná množina NLC pravidiel, C je binárna relácia spojení nad Σ a S je počiatočný graf.

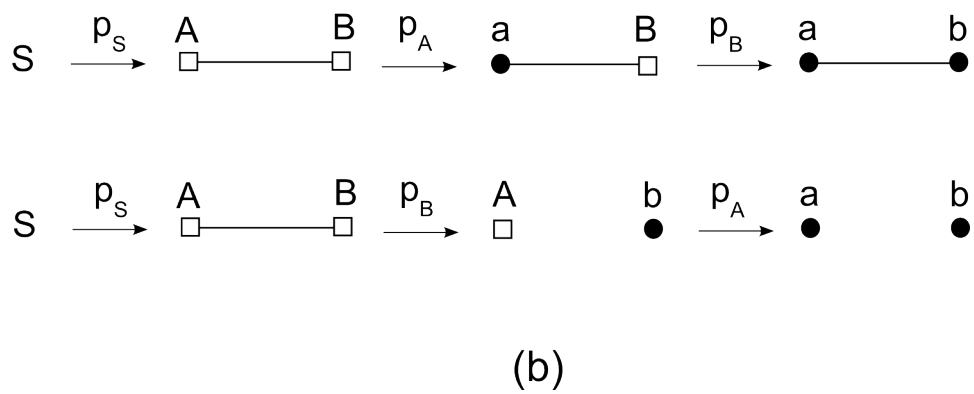
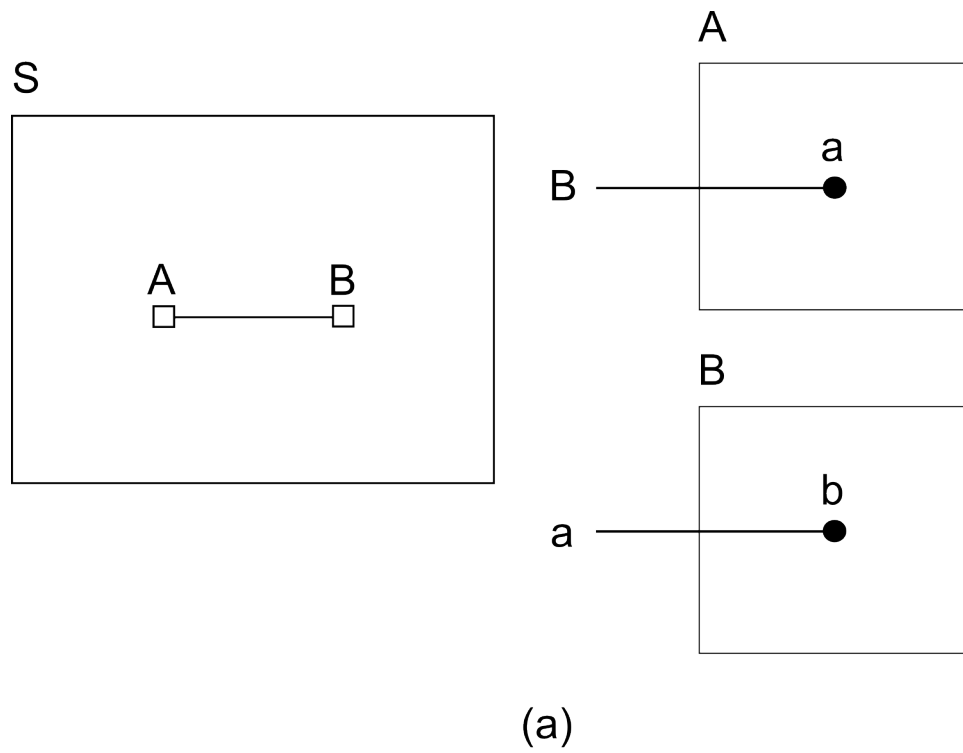
Definícia 2.1.2. *Grafový jazyk generovaný G je $L(G) = \{H \in GR_\Delta \mid S \Rightarrow^* H\}$, kde GR_Δ je množina neorientovaných grafov so značkami uzlov v Δ , $\Rightarrow$ reprezentuje jeden prepisovací krok, a $\Rightarrow^*$ reprezentuje odvodenie, t.j. postupnosť prepisovacích krokov.*

Príklad 2.1.2. Na obrázku 2.1 ilustrujeme pravidlá gramatiky z príkladu 2.1.1 spolu s relevantnými spájacími inštrukciami (použiteľnými pre dané pravidlo) a znázorňujeme jednu z možných reprezentácií jeho odvodení.

Poznámka 2.1.1. V pravidlách zložitejších tried gramatík a v ich odvodeniach sa budú značiť aj hrany.

2.2 NCE grafové gramatiky

NCE gramatiky tvoria rozšírenie NLC prepisovacieho systému. Dedia jeho vlastnosti ako atomickosť ľavej strany pravidla (prepisuje sa jeden uzol), lokálnosť prepisovania (pri prepisovaní sa dcérsky graf spojí len so susedmi prepisovaného uzla v materskom grafe) a ich pravidlá sú tiež bez podmienok na použitie. NCE mechanizmus prepisovania sa oproti NLC líši v tom, že množina spájacích inštrukcií C je definovaná pre každé pravidlo zvlášť a tým, že NCE gramatika dokáže pristupovať k jednotlivým uzlom D priamo (t.j. dokáže pristupovať ku každému uzlu dcérského grafu nezávisle na jeho značení, napríklad očíslovaním uzlov v pravidle). Existujú ďalšie rozšírenia prepisovacieho systému NCE, a to rozšírenie o značenie hrán a rozšírenie o ich orientáciu (značia sa písmenami e a d). edNCE gramatiky sú grafové gramatiky so susedskou kontrolou vnorenia (neighbourhood controlled embedding) a dynamickým preznačovaním hrán pri prepisovaní. Písmeno e znamená značenie hrán a d znamená že sa budú používať orientované grafy. Podobné rozšírenia sa dajú definovať aj pre NLC mechanizmus prepisovania. Pravidlo pre edNCE gramatiku má tvar $X \rightarrow (D, C)$. Vo všeobecnosti je inštrukcia z C tvaru $(m, \Pi/q, x, d)$, kde Π špecifikuje binárnu reláciu medzi uzlami nahradeného grafu M a zvyškom grafu H^- . Význam takejto inštrukcie je taký, že hrana so značkou q môže byť vytvorená medzi každým uzlom



Obrázok 2.1: (a) pravidlá gramatiky, (b) odvodenia gramatiky

y z H^- ktorý je v relácii Π s m zachovávajúc smer d (podobne môžeme prepisovať aj orientáciu hrany d/d').

Definícia 2.2.1. *edNCE grafová gramatika je definovaná ako 6-tica $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$. Kde Σ je abeceda značiek uzlov, $\Delta \subseteq \Sigma$ je abeceda terminálnych značiek uzlov, Γ je abeceda značiek hrán, $\Omega \subseteq \Gamma$ je abeceda terminálnych značiek hrán, P je konečná množina pravidiel a $S \in \Sigma - \Delta$ je počiatočný neterminál. Pravidlá sú tvaru $X \rightarrow (D, C)$, kde $X \in \Sigma - \Delta$ a $(D, C) \in GRE_{\Sigma, \Gamma}$. Relácia spojení je $C \subseteq \Sigma \times \Gamma \times \Gamma \times V_H \times \{in, out\}$.*

Poznámka 2.2.1. Graf považujeme za terminálny, ak všetky jeho uzly sú označené terminálmi (hrany nemusia byť označené terminálmi).

Vo všeobecnosti množinu grafov izomorfných s grafom H budeme značiť $[H]$. Niekedy sa hovorí o H ako „konkrétom“ grafe a o $[H]$ ako „abstraktnom“ grafe.

Definícia 2.2.2. *Množinu všetkých konkrétnych grafov nad Σ a Γ značíme $GR_{\Sigma, \Gamma}$ a množinu všetkých abstraktných grafov značíme $[GR_{\Sigma, \Gamma}]$. $GRE_{\Sigma, \Gamma}$ označuje množinu grafov s vnorením nad Σ, Γ . Obyčajný graf môžeme považovať za graf s prázdny vnorením, teda $GR_{\Sigma, \Gamma} \subseteq GRE_{\Sigma, \Gamma}$.*

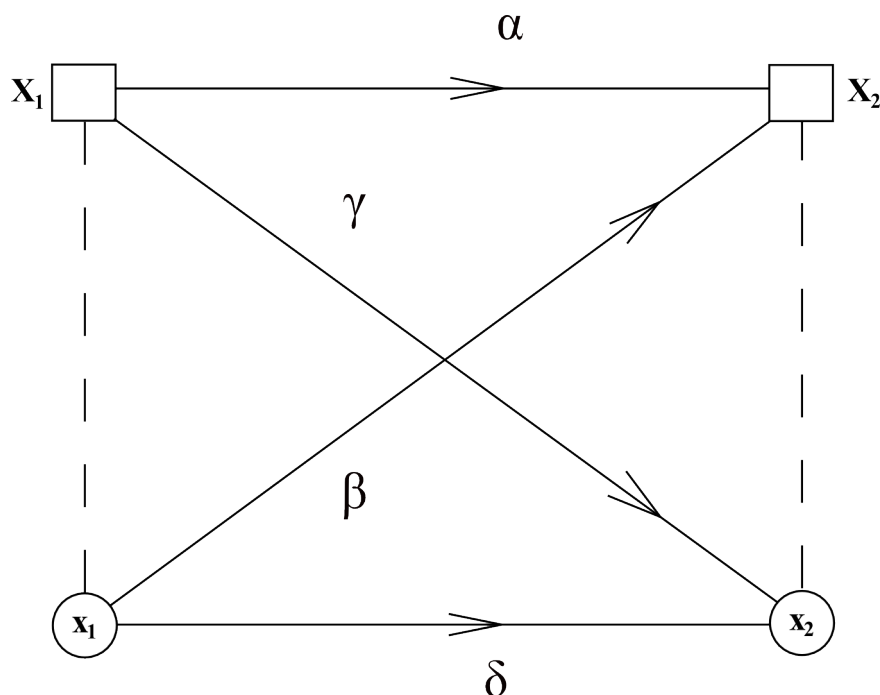
Definícia 2.2.3. *Grafový jazyk generovaný G je $L(G) = \{[H] \mid H \in GR_{\Delta, \Omega} \text{ a } sn(S, z) \Rightarrow^* H \text{ pre nejaké } z\}$ (t.j. všetky grafy v odvodení G , ktoré majú terminálne značky hrán aj uzlov).*

Definujeme teraz podtriedu edNCE gramatík, v ktorých zmena poradia použitých pravidiel nemení výsledný graf (ani vetnú formu odvodenia).

Poznámka 2.2.2. Na mysli máme také zmeny poradia pravidiel, ktoré umožňujú použitie pravidiel vo vetnej forme.

Definícia 2.2.4. *edNCE gramatika $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ je konfluentná (confluent), alebo C-edNCE gramatika, ak pre všetky pravidlá $X_1 \rightarrow (D_1, C_1)$ a $X_2 \rightarrow (D_2, C_2)$ z P , všetky uzly $x_1 \in V_{D_1}$ a $x_2 \in V_{D_2}$, a všetky hrany so značkami $\alpha, \delta \in \Gamma$, platia nasledujúce ekvivalencie:*

$$\exists \beta \in \Gamma : (X_2, \alpha/\beta, x_1, out) \in C_1 \text{ a } (\lambda_{D_1}(x_1), \beta/\delta, x_2, in) \in C_2$$



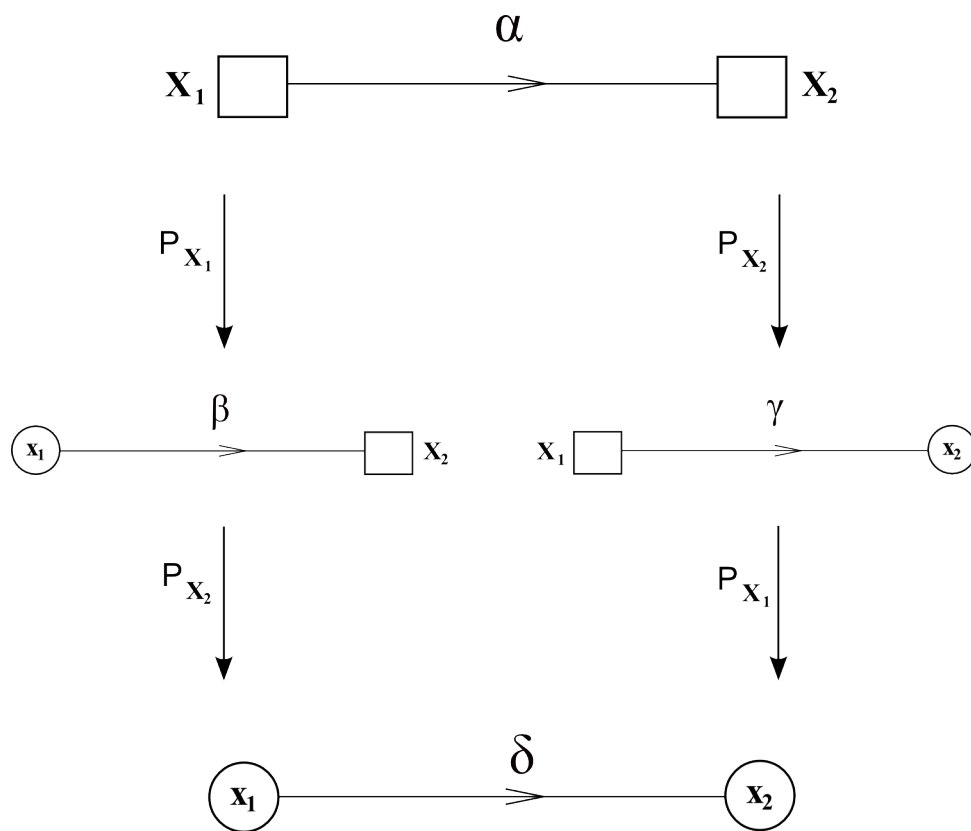
Obrázok 2.2: Konfluencia resp. Church Roserova vlasnosť

$$\Longleftrightarrow$$

$$\exists \gamma \in \Gamma : (X_1, \alpha/\gamma, x_2, in) \in C_2 \text{ a } (\lambda_{D_2}(x_2), \gamma/\delta, x_1, out) \in C_1.$$

Konfluencia je obmedzenie zaručujúce „pekné“ bezkontextové vlastnosti jazykov generovaných konfluentnými gramatikami. Ako už bolo spomenuté, táto vlastnosť síce bola zaručená v bezkontextových gramatikách pre reťazce, ale nie pre ich priamočiare zovšeobecnenie na grafové gramatiky. Podobným obmedzením je aj „boundary“, ktoré však viac znižuje mohutnosť množiny jazykov generovaných grafovými gramatikami s takýmto typom obmedzenia oproti gramatikám s obmedzením konfluencie. Existuje viacero definícií pre konfluenciu gramatiky. V podstate sú všetky rovnaké, čo sa týka generatívnej sily daného typu gramatík, ale nemusia a ani nie sú ekvivalentné v rámci jednotlivých gramatík.

Definícia konfluencie gramatiky popísaná na obrázku 2.2 a podrobnejšie na obrázku 2.3 sa dá voľne popísať nasledovne. Nech sú použité dve pravidlá na graf (konkrétna časť grafu) s jednou hranou (u_1, α, u_2) , kde u_i má značku, resp. je označený neterminálnym symbolom X_i . Potom po použití



Obrázok 2.3: Konfluencia

oboch pravidiel v akomkoľvek poradí vznikne tá istá hrana. V našom prípade je to (x_1, δ, x_2) . Z tejto definície intuitívne vyplýva spomenutá vlastnosť konfluencie gramatík, že výsledok odvodenia nezávisí na poradí použitých pravidiel.

Poznámka 2.2.3. Definícia konfluencie platí aj pre neterminálne uzly. Namísto terminálnych uzlov x_1ax_2 v definícii konfluencie môžu byť použité neterminálne uzly.

Poznámka 2.2.4. Obrázky 2.2 a 2.3 konfluentnej gramatiky znázorňujú len konkrétnu dvojicu uzlov spojených hranov, ktoré sú ovplyvnené pravidlom gramatiky. Pravidlo vo všeobecnosti môže ovplyvňovať aj iné časti vetnej formy.

Príklady grafov generovaných C-edNCE gramatikami sú stromy, (Kvôli jednoznačnosti uvádzame grafové jazyky v anglickej terminológii.) cographs, series-parallel graph, transitive VSP graphs, uniformly non-primitive 2-structures, complete bipartite graphs, acyclic tournaments, (maximal) outerplanar graphs, edge complements of trees, a pre fixné k , k -trees, graphs of treewidth $\leq k$, pathwidth $\leq k$, cutwidth $\leq k$, bandwidth $\leq k$, cyclic bandwidth $\leq k$, topological bandwidth $\leq k$. [7]

Definícia 2.2.5. edNCE gramatiku G nazývame neblokujúca, ak každá terminálna vetná forma H v G má len hrany označené terminálom, t.j. finálne hrany.

2.3 Ďalšie triedy grafových gramatík

Definícia 2.3.1. Gramatika $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ je ohraničená (boundary), alebo B-edNCE, ak pre každé pravidlo $X \rightarrow (D, C)$ platí:

1. D neobsahuje hranu medzi dvoma uzlami označenými neterminálom
2. C neobsahuje inštrukciu spojenia $(\sigma, \beta/\gamma, x, d)$, kde σ je neterminál.

Definícia 2.3.2. B-edNCE gramatika $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ je nonterminal neighbour deterministic, alebo B_{nd} -edNCE, ak pre každé pravidlo $X \rightarrow$

(D, C) , každý neterminálny uzol $x \in V_D$, všetky $\gamma \in \Gamma$ a $\sigma \in \Delta$ platí:

1. $\{y \in V_D \mid (y, \gamma, x) \in E_D \text{ a } \lambda_D(y) = \sigma\} \cup \{\beta \in \Gamma \mid (\sigma, \beta/\gamma, x, in) \in C\}$ je singleton alebo prázdne.
2. $\{y \in V_D \mid (x, \gamma, y) \in E_D \text{ a } \lambda_D(y) = \sigma\} \cup \{\beta \in \Gamma \mid (\sigma, \beta/\gamma, x, out) \in C\}$ je singleton alebo prázdne.

Predpokladáme, že $V_D \cap \Gamma = \emptyset$.

Definícia 2.3.3. *C-edNCE gramatika má ohraničený neterminálny stupeň alebo C_{bntd} -edNCE gramatika je taká, ktorá má ohraničený stupeň neterminálnych uzlov vo vetnej forme.*

Definícia 2.3.4. *edNCE gramatika $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ je apex alebo A-edNCE gramatika, ak pre každé pravidlo $X \rightarrow (D, C)$ a každú inštrukciu spojenia $(\sigma, \beta/\gamma, x, d) \in C$, x a σ sú terminály.*

Definícia 2.3.5. *edNCE gramatika $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ je lineárna alebo LIN-edNCE gramatika, ak pre každé pravidlo $X \rightarrow (D, C)$, D má najviac jeden neterminálny uzol.*

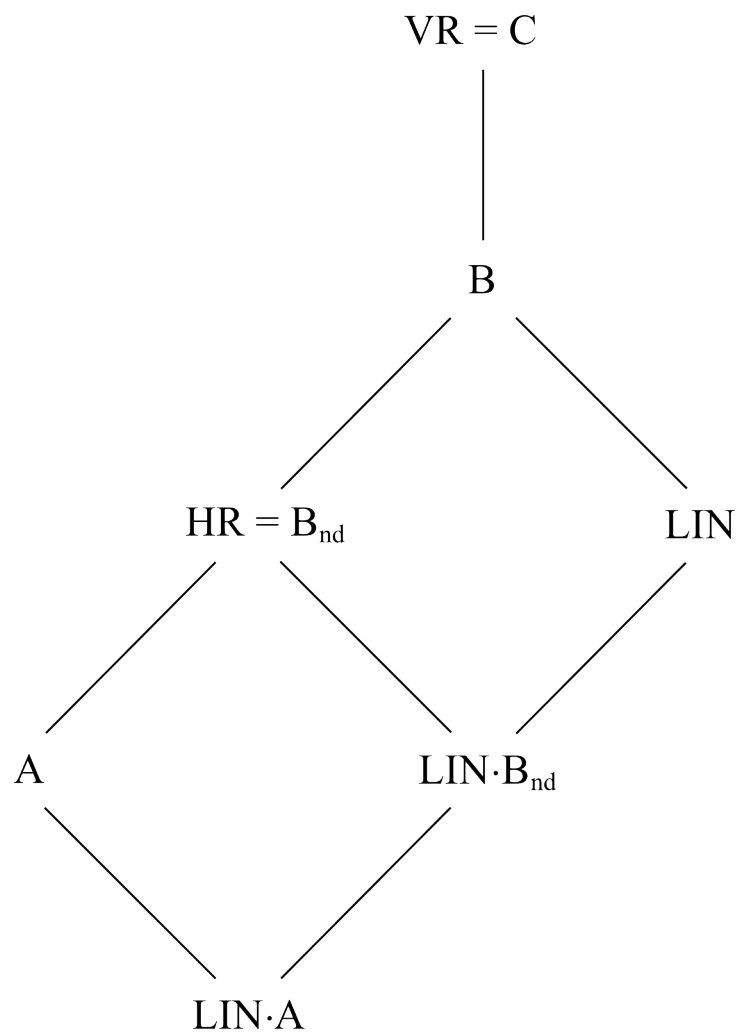
Hierarchia jazykov generovaných gramatikami, ktoré sme si tu definovali je zobrazená na obrázku 2.4.

2.4 Normálne tvary C-edNCE gramatík

Podobne ako v prípade gramatík generujúcich slová je často užitočné používať grafové gramatiky, v ktorých platia nejaké ohraničenia. Popíšeme teraz niektoré takéto obmedzenia pre C-edNCE gramatiky, ktoré neznižujú ich generatívnu silu (t.j., triedu C-edNCE gramatík s príslušným ohraničením môžeme považovať za gramatiky v normálnom tvare). Príkladmi normálnych tvarov, podobne ako pre jazyky nad reťazcami sú, odstránenie nedosiahnuteľných, neterminálniteľných neterminálov, alebo gramatiky bez Λ či chain pravidiel.

Normálnym tvarom je aj trieda L-edNCE označujúca gramatiky edNCE, v ktorých používame len najľavejšie odvodenie.

Veta 2.4.1. $L\text{-edNCE} = C\text{-edNCE}$



Obrázok 2.4: Hierarchia edNCE jazykov.

Táto veta hovorí o tom, že najľavejšie odvodenie v edNCE gramatike je ekvivalentné odvodeniu v nejakej C-edNCE gramatike. Myšlienka spočíva v tom, že keď si očísľujeme uzly vo vetnej forme, tak nám budú tvoriť jedinečný spôsob prechádzania stromu odvodenia. [7]

Jedna z techník, ktoré sa používajú na „zablokovanie“ nežiaducich odvodení v gramatike je použitie tzv. blokujúcich hrán. Sú to hrany označené neterminálnym symbolom, ktorý sa už nedá prepísať na terminálny. Z takéhoto grafu už odvodenie nemôže pokračovať tak, aby vznikol graf patriaci do jazyka tejto gramatiky.

Definícia 2.4.1. *Blokujúca hrana je hrana medzi dvoma terminálnymi uzlami, ktorá je označená neterminálnym symbolom, t.j. nefinálna hrana, ktorá sa už nedá prepísať na terminálnu alebo odstrániť.*

Za normálny tvar C-edNCE gramatík môžeme považovať aj C-edNCE gramatiky bez blokujúcich hrán.

Veta 2.4.2. Ku každej C-edNCE gramatike vieme zostrojiť ekvivalentnú C-edNCE gramatiku bez blokujúcich hrán. [7]

Vďaka tejto vete môžeme rozhodnúť problém prázdnoty a konečnosti C-edNCE rovnakým spôsobom ako pre bezkontextové jazyky nad reťazcami.

Veta 2.4.3. Pre neblokujúcu grafovú gramatiku G , reťazcový jazyk: $\{a^n \mid n = \#V_H \text{ pre nejaký } [H] \in L(G)\}$ je bezkontextový. [7]

Dôkaz. Každé pravidlo $X \rightarrow (D, C)$, ktorého uzly sú označené $\sigma_1, \dots, \sigma_k$, prepíšeme na pravidlo $X \rightarrow \sigma'_1 \dots \sigma'_k$ kde $\sigma'_i = a$, ak $\sigma_i \in \Delta$, inak $\sigma'_i = \sigma_i$. $\square$

Veta hovorí o bezkontextových vlastnostiach edNCE prepisovacieho systému pokiaľ neuvažujeme štruktúru generovaných grafov. Táto totiž môže do jazyka vnášať nebezkontextové vlastnosti. Napríklad vďaka blokujúcim hranám môžu vznikať kontextové reťazcové jazyky. Alebo kvôli rôznym množinám spájacích inštrukcií pre jednotlivé pravidlá, sa nemusia uzly a hrany daného značenia spájať (uzly) a prepisovať (hrany) vždy rovnakým spôsobom ako to bolo uvedené v príklade 2.1.1.

Ako dôsledok vety 2.4.2 dostávame nasledujúce tvrdenie, ktoré nám často pomôže ukázať, že nejaký grafový jazyk sa nedá generovať C-edNCE gramatikou. Spresnené znenie a dôkaz uvedieme medzi vlastnými výsledkami v Kapitole 5.

Veta 2.4.4. [Veta o medzere] Nech L je C-edNCE grafový jazyk a $n_1, n_2, n_3, \dots$ je usporiadaná postupnosť čísel obsahujúca číslo k práve vtedy, ak L obsahuje graf s k uzlami. Potom existuje konštanta p taká, že pre každé i platí $(n_{i+1} - n_i) \leq p$.

Dôkaz. Plyní z vety 2.4.3 a pumpovacej lemy pre bezkontextové jazyky nad reťazcami. Spresnený dôkaz a znenie vety uvádzame v Kapitole 5. $\square$

Definícia 2.4.2. Λ -pravidlo je tvaru $X \rightarrow \emptyset$. Chain pravidlo je tvaru $X \rightarrow Y$, kde Y je jeden neterminálny uzol.

Veta 2.4.5. Ku každej C-edNCE gramatike vieme zostrojiť ekvivalentnú C-edNCE gramatiku bez Λ -pravidiel a chain pravidiel. Rovnaké tvrdenia platia pre podtriedy X-edNCE, X^*Y -edNCE gramatiky sú edNCE gramatiky pre ktoré platí X aj Y . Podtriedy: B - boundary, A - apex, LIN - lineárne, ...

Konštrukcie sú uvedené v [7] a sú analogické ako pre bezkontextové jazyky. Navyše je len výpočet inštrukcie spojenia pre chain pravidlo.

Poznámka 2.4.1. Pre odstránenie Λ -pravidiel v edNCE gramatike funguje rovnaká konštrukcia.

Chomského normálny tvar

Analogicky ako pre jazyky nad reťazcami, existuje aj pre C-edNCE jazyky Chomského normálny tvar (CHNT).

Veta 2.4.6. Ku každej C-edNCE gramatike vieme skonštruovať ekvivalentnú C-edNCE gramatiku v CHNT. Rovnaké tvrdenie platí aj pre podtriedy B , B_{nd} , LIN , $LIN * B_{nd}$. Pre triedu *apex* neexistuje CHNT. [7]

Operátorový normálny tvar

ONF - operátorová normálna forma: pravá strana každého pravidla obsahuje

práve jeden terminálny uzol. V každom kroku odvodenia vznikne práve jeden terminál. Zatiaľ nepoznáme dôkaz existencie ekvivalentnej C-edNCE gramatiky v ONF pre každú C-edNCE gramatiku. Pre každú B-edNCE gramatiku existuje B-edNCE v ONF. Podobne pre A-edNCE. Každá LIN-edNCE gramatika v CHNT je aj v ONF.

Normálne tvary grafových gramatík a jazykov sa podobne ako pre reťazce využívajú v dôkazoch zložitosti rozpoznávania, charakterizácie a podobne.

2.5 Charakterizácia C-edNCE triedy jazykov

Trieda bezkontextových grafových jazykov C-edNCE môže byť charakterizovaná mnohými spôsobmi. Medzi najčastejšie patria: najľavejšie odvodenia v edNCE, pomocou stromov a reťazcových jazykov, MSO charakterizácia a HR gramatiky.

2.5.1 Logická charakterizácia

Najužitočnejšia a na gramatikách úplne nezávislá charakterizácia C-edNCE je pomocou Monadic Second Order(MSO) logiky. Charakterizácia hovorí o tom, že C-edNCE je trieda všetkých grafových jazykov $f(T_\Delta)$, kde T_Δ je množina stromov nad abecedou terminálnych uzlov Δ a f je MSO definovateľná funkcia zo stromov na grafy.

- f je definovaná na stromoch, preto grafy $f(t)$ sú „tree-like“.
- MSO logika je vhodný jazyk pre popis grafov, môže byť použitý ako špecifikačný jazyk pre množiny grafov a funkcií na grafoch.
- Veľa výsledkov pre C-edNCE jazyky má dôkazy využívajúce MSO charakterizáciu, ktorá je oveľa jednoduchšia ako charakterizácia pomocou C-edNCE gramatík.

Definícia 2.5.1. $MSOL(\Sigma, \Gamma)$ - MSO jazyk nad Σ, Γ sa skladá:

- z premenných uzlov - $u, v, \dots$

- z premenných množín uzlov - $U, V, \dots$
- zo štyroch typov atomických formúl (pre každé $\sigma \in \Sigma$ a každé $\gamma \in \Gamma$):
 - $lab_\sigma(u)$ - uzol u má značku σ
 - $edge_\gamma(u, v)$ - existuje hrana so značkou γ z u do v
 - $u = v$
 - $u \in U$

Formule jazyka MSO sú potom zostrojené z atomických formúl pomocou propozičných spojok $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$ a kvantifikátorov $\forall$ a $\exists$. Kvantifikované môžu byť len premenné uzlov a premenné množín uzlov.

Grafový jazyk definovaný pomocou MSO formule je v príklade 2.5.1.

Definícia 2.5.2 (Uzavretosť formuly). *Formula je uzavretá, ak neobsahuje žiadny voľný výskyt premennej.*

Definícia 2.5.3. *Grafový jazyk $L \subseteq [GR_{\Sigma, \Gamma}]$ je MSO definovateľný, ak existuje uzavretá formula ϕ v $MSOL(\Sigma, \Gamma)$ taká, že*

$$L = \{[H] \in [GR_{\Sigma, \Gamma}] \mid H \models \phi\}$$

t.j., L je množina všetkých grafov v $[GR_{\Sigma, \Gamma}]$ pre ktoré platí ϕ .

Príklad 2.5.1. Formula pre bipartitu grafu:

$$\exists U \exists V : (part(U, V) \wedge \forall u \forall v : edge(u, v) \rightarrow (u \in U \wedge v \in V) \vee (u \in V \wedge v \in U))$$

, kde $part(U, V)$ je pravdivá ak U, V tvorí partie uzlov grafu.

Definícia 2.5.4. *Nech $\Sigma_i \Gamma_i$ sú abecedy, pre $i \in \{1, 2\}$. MSO definovateľná funkcia f :*

$$GR_{\Sigma_1, \Gamma_1} \rightarrow GR_{\Sigma_2, \Gamma_2}$$

je daná nasledovnými formulami v $MSOL(\Sigma_1, \Gamma_1)$:

- uzavretou formulou ϕ_{dom} , doménová formula

- formula $\phi_\sigma(u)$, pre každé $\sigma \in \Sigma_2$, formula pre uzly
- formula $\phi_\gamma(u, v)$, pre každé $\gamma \in \Gamma_2$, formula pre hrany

Doména pre f je $\{H \in GR_{\Sigma_1, \Gamma_1} \mid H \models \phi_{dom}\}$, a pre každé $H \in dom(f)$, $f(H)$ je graf $(V, E, \lambda) \in GR_{\Sigma_2, \Gamma_2}$ taký, že

- $V = \{x \in V_H \mid \text{existuje práve jedna } \sigma \in \Sigma_2 \text{ taká, že } H \models \phi_\sigma(x)\}$
- $E = \{(x, \gamma, y) \mid x, y \in V, x \neq y, \gamma \in \Gamma_2 \text{ a } H \models \phi_\gamma(x, y)\}$
- pre $x \in V$, $\lambda(x) = \sigma$ kde $H \models \phi_\sigma(x)$

Príklady MSO definovateľných funkcií: preznačenie uzlov, preznačenie hrán, hranový komplement, tranzitívny uzáver, a podobne.

Veta 2.5.1. C-edNCE a LIN-edNCE sú uzavreté na MSO definovateľné funkcie. Inak povedané sú uzavreté na prienik s MSO definovateľnými množinami grafov. [7]

2.5.2 Príklady využitia MSO charakterizácie

Jedným z príkladov využitia MSO charakterizácie sú *Slisenkove gramatiky* ([12]). Slisenkove gramatiky generujú také triedy grafov, pre ktoré je existencia hamiltonovej kružnice rozhodnuteľná v polynomiálnom čase. Využívajú vlastnosť triedy grafov s ohraničenou šírkou stromu (bounded tree-width). Nad triedami grafov s ohraničenou šírkou stromu sú vlastnosti definovateľné v MSOL rozpoznateľné v polynomiálnom čase. Hamiltonova kružnica je definovateľná pomocou MSOL.

Ak K je M-uzavretá trieda grafov, potom K má ohraničenú šírku stromu, práve vtedy, keď každá MSO vlastnosť je rozhodnuteľná v P . Ak $P \neq NP$ a v triede grafov K , každá existenčná MSO definovateľná vlastnosť je v P , potom K má ohraničenú šírku stromu. Ak K je uzavretá na minory, potom K je v P . [6]

Veta 2.5.2. Pre každú bezkontextovú grafovú gramatiku a každú vlastnosť grafu Π , ktorú vieme sformulovať v MSOL je rozhodnuteľné, či existuje graf

generovaný gramatikou spĺňajúci II. Výsledok je aplikovateľný na HR a C-NLC grafové gramatiky. [3, 7]

Nasledujúca veta poskytuje abstraktnejší pohľad na vlastnosti a podobu C-edNCE grafových jazykov. Takisto je aj akýmsi zhrnutím predchádzajúcich poznatkov a dôsledkov, pričom popisuje rôzny charakter tried grafových jazykov generovaných edNCE a C-edNCE gramatikami. Veta hovorí o tom, že každý C-edNCE jazyk obsahuje len také nekonečné MSO definovateľné jazyky, pre ktoré platí veta 2.4.4.

Veta 2.5.3. Pre každú MSO definovateľnú množinu grafov, ktorá tvorí nekonečnú podpostupnosť nejakého bezkontextového grafového jazyka z C-edNCE platí veta 2.4.4. Pre edNCE rovnaké tvrdenie neplatí.

Dôkaz. Z $L(G) \in C - edNCE$ a z uzavretosti C-edNCE jazykov na prienik s MSO definovateľnými triedami grafov dostávame, že daná podpostupnosť musí byť tiež bezkontextová a preto pre ňu platí veta 2.4.4.

Pre jazyky generované neblokujúcimi edNCE gramatikami podobné tvrdenie neplatí. Ako dôkaz uvedieme kontra príklad, resp. vlastnosť neuzavretosti neblokujúcich edNCE jazykov na prienik s MSO definovateľnými triedami grafov. Uvažujme jazyk $[GR_{\Delta, \Omega}]$. V tomto jazyku existuje nekonečná MSO definovateľná podpostupnosť napr. $GRID_{n,n}$ - postupnosť štvorcových mriežok. Ak by jazyky generované neblokujúcimi edNCE gramatikami boli uzavreté na prienik s MSO definovateľnými triedami grafov, potom pre túto postupnosť musí platiť veta 2.4.4. Čo očividne neplatí a preto tieto jazyky nie sú uzavreté na prienik s MSO definovateľnými triedami grafov. $\square$

Ako príklad využitia tejto vety dostávame nasledujúci dôsledok.

Dôsledok 2.5.1. Žiadny C-edNCE grafový jazyk neobsahuje nekonečne veľa štvorcových mriežok.

Dôkaz. Množina *všetkých mriežok* je MSO definovateľná: mriežka je acyklický súvislý graf, pre ktorý platí: každý uzol ma najviac jednu vychádzajúcu resp. vchádzajúcu vertikálnu resp. horizontálnu hranu, ak (x, h, y_1) ,

(y_1, v, z_1) , (x, v, y_2) a (y_2, h, z_2) , potom $z_1 = z_2$, ak existuje horizontálna hrana z x do y alebo z y do x a x má vychádzajúcu resp. vchádzajúcu vertikálnu hranu, potom aj y . Podobne pre vertikálnu hranu z x do y resp. z y do x ak x má horizontálnu vchádzajúcu resp. vychádzajúcu hranu, potom aj y . Množina *štvorcových mriežok* je tiež MSO definovateľná: mriežka je štvorec ak v nej existuje cesta π z uzla bez vchádzajúcej hrany do uzla bez vychádzajúcej hrany taká, že postupnosť značiek hrán cesty π je v $(hv)^*$ - h je horizontálna a v je vertikálna hrana (π je diagonálna). Podľa vety 2.4.3 a vety 2.5.1 žiadny C-edNCE jazyk neobsahuje nekonečný jazyk štvorcových mriežok. $\square$

Spomenutá veta 2.5.3 tvorí zovšeobecnenie a zhrnutie predchádzajúcich výsledkov. Poskytuje spôsob odlíšenia nebezkontextových jazykov od triedy C-edNCE, resp. ne-MSO definovateľných nekonečných podpostupností grafov od MSO definovateľných. Jej použitie je však podmienené nájdením vhodného kontra príkladu. Ak jazyk obsahuje nekonečnú MSO definovateľnú podpostupnosť grafov pre ktorú neplatí veta 2.4.4, potom tento jazyk nepatrí do C-edNCE triedy jazykov. Tento spôsob odlíšenia nie je veľmi efektívny, lebo nepopisuje žiadnu konkrétnu vlastnosť nebezkontextových jazykov. Inak povedané, je ťažké nájsť nekonečnú MSO definovateľnú podpostupnosť, ktorá by dokazovala, že daný jazyk nepatrí do C-edNCE triedy jazykov. Takisto rieši, i keď tiež neefektívne, problém MSO definovateľnosti nekonečnej podpostupnosti vyskytujúcej sa v C-edNCE jazyku. Ak daný jazyk patrí do C-edNCE, potom žiadna nekonečná podpostupnosť grafov z tohto jazyka, pre ktorú neplatí veta 2.4.4, nie je MSO definovateľná.

Poznámka 2.5.1. Obsiahnutie nekonečných MSO definovateľných grafových jazykov, pre ktoré platí resp. neplatí veta 2.4.4, slúži na odlišovanie medzi C-edNCE jazykmi a jazykmi ktoré umožňuje generovať edNCE prepisovací systém a jeho rozšírenia resp. MSO definovateľnými a MSO nedefinovateľnými nekonečnými postupnosťami.

Okrem iného aj podľa vyššie uvedenej vety nie sú nasledujúce jazyky bezkontextové: všetky grafy stupňa ≤ 4 , planárne grafy, súvislé grafy, všetky grafy stupňa $\geq 2,3$ a 4. Obsahujú napríklad nekonečnú MSO definovateľnú

postupnosť štvorcových mriežok, ktorá ako už bolo spomenuté nie je bezkontextová resp. pre ňu neplatí veta 2.4.4. Podobne by sme mohli uviesť aj ďalšie príklady jazykov.

2.6 Uzáverové vlastnosti bezkontextových grafových jazykov

Väčšina uzáverových vlastností C-edNCE triedy jazykov vyplýva z uzavretosti na prienik s MSO definovateľnými množinami grafov a z uzavretosti na MSO definovateľné funkcie. Na rozdiel od NLC jazykov, ktoré nie sú uzavreté napríklad na preznačenie uzlov, čo je MSO definovateľná funkcia ako sme už spomenuli vyššie. Niektoré uzáverové vlastnosti sa dajú ukázať podobne ako pre bezkontextové jazyky nad reťazcami.

Napríklad zjednotenie ($\sigma \rightarrow \sigma_1 \mid \sigma_2$)

Kapitola 3

Regulárne grafové gramatiky a ich normálny tvar

V tejto kapitole prinášame vlastnú, jednoduchú konštrukciu normálneho tvaru pre regulárne grafové gramatiky. Vychádzame z analógie s regulárnymi gramatikami pre reťazce.

Definícia 3.0.1. *Nazvime regulárnymi grafovými gramatikami tie grafové gramatiky, ktoré majú tvar pravidla definovaný, analogicky ako regulárne gramatiky pre reťazce, nasledovne: pravidlo má tvar $N \rightarrow (NT^*, C)$ alebo $N \rightarrow (T^*, C)$.*

Ak obmedzíme počet neterminálov vo vetnej forme edNCE gramatiky, napríklad spôsobom z predošlej definície, tak nám to pre tieto gramatiky umožní zdefinovať nasledujúci normálny tvar.

Veta 3.0.1. Ku každej edNCE gramatike G_1 s regulárnymi pravidlami existuje ekvivalentná edNCE gramatika G_2 v takom normálnom tvare, kde všetky pravidlá sú tvaru $X \rightarrow (D, C)$, kde D obsahuje najviac jeden neterminálny uzol a práve jeden terminálny uzol.

Dôkaz. Uvažujme gramatiku bez Λ a chain pravidiel. Podľa poznámky k vete 2.4.5 a obmedzenia počtu neterminálov vo vetnej forme si to môžeme dovoliť. Podobne, nech je táto gramatika bez nedosiahnuteľných a neterminálniteľných pravidiel.

Všetky pravidlá G_1 tvaru $X \rightarrow (D, C)$, kde D sa skladá z viac ako jedného terminálneho uzla, v G_2 nahradíme novými pravidlami nasledovne:

- Ak je na pravej strane viac ako jeden terminálny uzol, tak pravidlo rozdelíme podľa štandardnej konštrukcie uvedenej pre jazyky nad reťazcami nasledovne. Každé pravidlo tvaru $X \rightarrow T^i N$ resp. $X \rightarrow T^i$, kde $i > 1$, resp. $i > 2$ prepíšeme na nové pravidlá $X \rightarrow T_1 \xi_1, \dots, \xi_{i-2} \rightarrow T_{i-1} \xi_{i-1}$, resp. podobne posledné pravidlo v druhom prípade bude $\xi_{i-1} \rightarrow T_i$. Pričom všetky terminály z dcérskeho grafu D prepisovaného „starého“ pravidla budú pri vytváraní D spojené s neterminálmi ξ z jednotlivých pravých strán „nových“ pravidiel, čo zaručuje postupné generovanie grafu pravej strany pôvodného pravidla pomocou inštrukcií spojenia „nových“ pravidiel. Každé „nové“ pravidlo spája novovytvorený terminálny uzol len s tými uzlami s ktorými bol spojený v dcérskom grafe pravej strany pôvodného pravidla (postupne vytvára graf D pôvodného pravidla). Toto postupné generovanie grafu D vieme zaručiť vďaka konečnej pravej strane „starého“ pravidla pomocou vhodne zvolených značení hrán. Pre jednoduchosť budeme uvažovať, že pre každé spojenie neterminálu s terminálom (pri vytváraní D) budeme mať hranu s jedinečným značením, ktoré bude určovať, ktorý terminálny uzol (spojený s aktuálne prepisovaným neterminálom) sa má spojiť s novovzniknutým terminálom a ktorý nie („napevno“ zapíšeme do sady spájacích inštrukcií daného pravidla). Na konci sa neterminál prepíše na terminál, alebo na terminál a neterminál pričom sa spoja podľa pôvodného pravidla len s tými terminálnymi uzlami s ktorými boli spojené aj v pôvodnom pravidle, čo nám opäť zaručia jedinečné hrany medzi existujúcimi terminálmi a prepisovaným neterminálom. Zároveň sa posledným pravidlom budú prepisovať novo definované jedinečné hrany na hrany definované podľa starého pravidla. K zvyšku grafu sa vytváraný graf pripája rovnako, čiže podľa inštrukcií spojenia starého pravidla. Nové značenia hrán pri vytváraní D sme použili, aby sme dokázali zabezpečiť spojenie vytváraného terminálu s uzlami D a pritom prípadne vedeli zamedziť prepojeniu s terminálmi zo zvyšku vetnej formy, ktoré majú rovnaké značenie ako uzly (hranu budú mať určite

rôznu), ktoré sú vytvárané v D . Touto konštrukciou sme tiež zabezpečili odlíšenie jednotlivých uzlov (s rovnakými značkami) vznikajúceho dcérskeho grafu.

- v ostatných prípadoch pravidlo zostane rovnaké

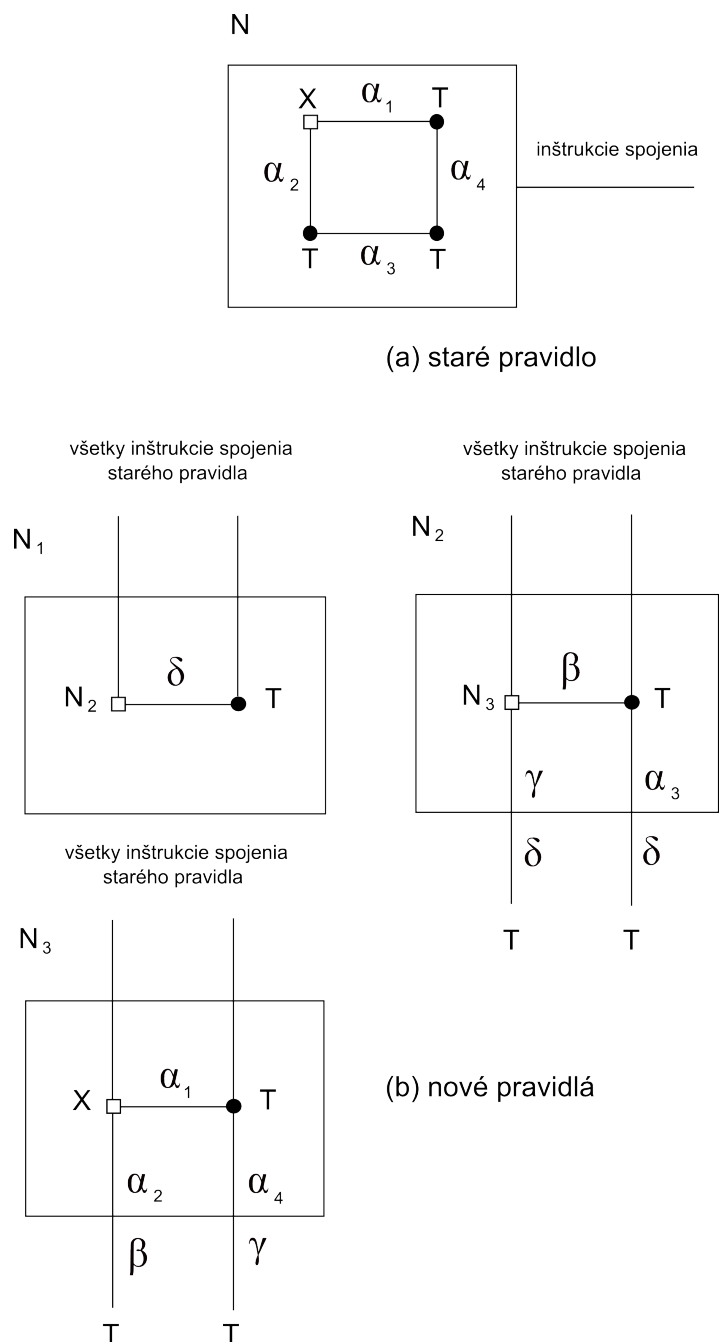
Z uvedenej konštrukcie vyplýva platnosť vety.

□

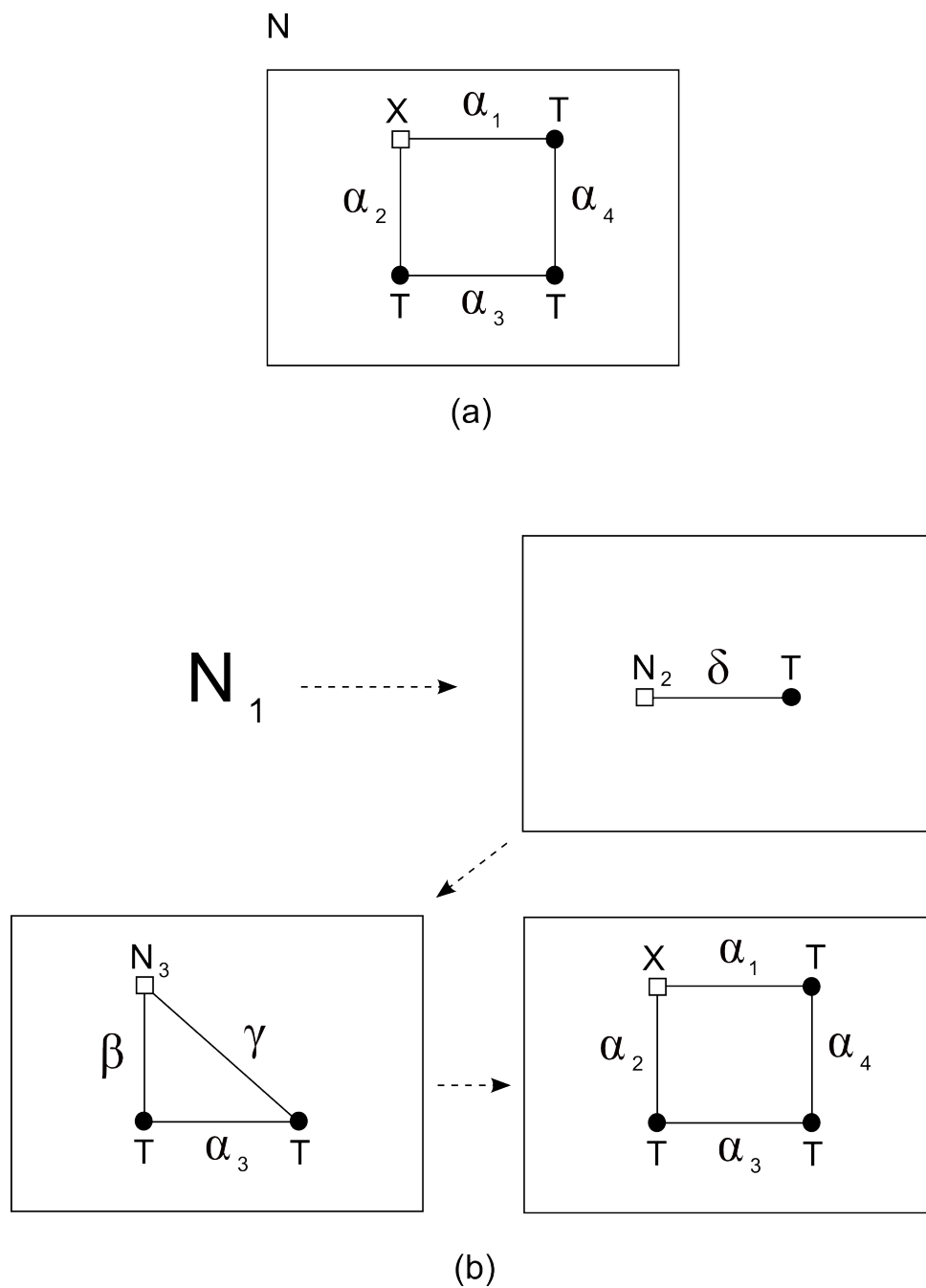
Prevod pravidiel gramatiky na tento normálny tvar a ich použitie v odvodení je zobrazené na ilustračných obrázkoch a 3.1 a 3.2.

Ľahko sa dá nahliadnuť, že podobné tvrdenie pre edNCE gramatiky vo všeobecnosti neplatí. Nekonfluentné gramatiky sú jednoduchým príkladom.

Veta 3.0.1 naznačuje obmedzené možnosti, regulárnej grafovej gramatiky, využívať hrany vo vetnej forme pri generovaní výsledného grafu. Z čoho vo všeobecnosti vyplýva väčšia zložitosť pravidiel regulárnej grafovej gramatiky oproti gramatike bez tohto obmedzenia. Z časti, i keď nepriamo, ukazuje pohľad na využitie blokujúcej hrany vo vetnej forme odvodenia, v podobe prenosu informácie medzi dvoma neterminálnymi uzlami. Uvedený systém prepisovania, ako sa dozvieme neskorších výsledkov a poznatkov, má značne oslabené generatívne schopnosti. Okrem iného, napríklad v dôsledku toho, že uvedené obmedzenie neumožňuje využívať kontextovú informáciu prenášanú pomocou blokujúcej hrany.



Obrázok 3.1: Ilustračný obrázok starého pravidla a nových pravidiel. (a) staré pravidlo, (b) nové pravidlá.



Obrázok 3.2: Ilustračný obrázok prepisu starého pravidla novými pravidlami.
 (a) staré pravidlo, (b) prepis novými pravidlami.

Kapitola 4

Kvantitatívne vlastnosti grafov a vety o medzere

V tejto kapitole prinášame výsledky našej snahy porozumieť vlastnostiam edNCE jazykov. Uvádzame niekoľko tvrdení typu „veta o medzere“ pre kvantitatívne parametre grafov v jazykoch generovaných edNCE gramatikami (počty uzlov, počty hrán a stupeň uzla). Najprv uvedieme spresnenú vetu o medzere pre počty uzlov grafov v C-edNCE jazykoch aj s dôkazom. V ďalších dôkazoch uvádzame menej detailov a odvolávame sa na analógiu úvah. V závere kapitoly uvedieme naše pozorovania o tom prečo je skúmanie kvantitatívnych vlastností edNCE jazykov ťažké.

Najskôr uvedieme pojem stromu odvodenia pre grafové pre grafové gramatiky, ktorý nám uľahčí argumentáciu v dôkazoch.

Strom odvodenia pre edNCE gramatiky:

Vďaka možnosti priameho prístupu, resp. priamemu prístupu k jednotlivým uzlom dcérskeho grafu pri odvodení, môžeme na pravej strane pravidla edNCE gramatik definovať usporiadanie uzlov. Preto strom odvodenia môžeme definovať rovnakým spôsobom ako pre bezkontextové jazyky nad reťazcami.

Definícia 4.0.2. *Strom odvodenia vetnej formy edNCE gramatiky je zakorenený a usporiadaný strom, v ktorom každý uzol je spojený orientovanými*

hranami so svojimi k synmi, $k \geq 0$, a poradie synov daného uzla je určené očíslovaním týchto hrán. Malý rozdiel oproti stromom odvodenia pre jazyky nad reťazcami je v tom, že uzly sú značené pravidlami (p -strom odvodenia), alebo kópiami pravidiel (c -strom odvodenia).

Poznámka 4.0.1. Množina kópií pravidla, značenie $copy(P)$, je množina všetkých pravidiel, ktoré sú izomorfné s pravidlom v P . Prvok tejto množiny sa nazýva kópia pravidla. My budeme vždy uvažovať, že P neobsahuje izomorfné pravidlá. Dve pravidlá sú izomorfné, ak majú na svojich ľavých stranách rovnaký neterminál a ich pravé strany sú izomorfné grafy.

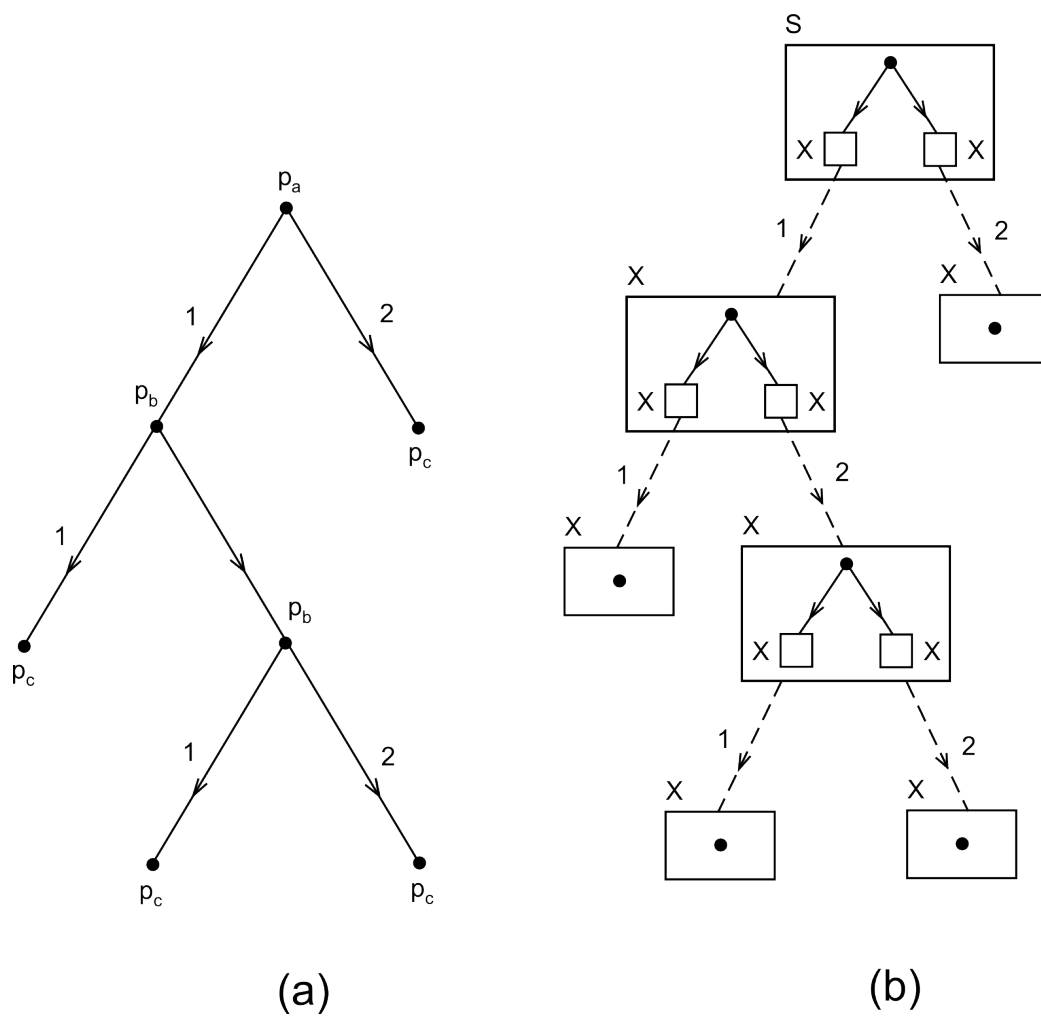
4.1 Vlastnosti grafov generovaných grafovými gramatikami

Odlíšenie jednotlivých tried grafových jazykov je v porovnaní s gramatikami a jazykmi nad reťazcami náročnejšie. Čiže je oveľa ťažšie nájsť a definovať hierarchiu pre grafové jazyky podobnú Chomského hierarchii. Táto kapitola je pokusom o porovnanie a odlíšenie existujúcich hierarchií grafových jazykov, ktoré vznikli vyžadovaním konfluencie, resp. odstránením blokujúcich hrán v edNCE prepisovacom systéme.

Teraz uvedieme spresnené znenie vety 2.4.4 o medzere z Kapitoly 3 aj s dôkazom.

Veta 4.1.1. [Veta o medzere] Nech jazyk $L(G) \in \text{C-edNCE}$, P nech je postupnosť všetkých slov z $L(G)$, usporiadaná podľa počtu uzlov v slovách. Potom pre každú dvojicu a, b , bezprostredne po sebe nasledujúcich slov z P platí: $||V_b| - |V_a|| \leq \max(rhs(p))^{||\Sigma| - |\Delta| + 1}$, kde V_a a V_b sú množiny uzlov pre a, b a $\max(rhs(p))$ je maximálny počet uzlov pravej strany pravidiel gramatiky G .

Myšlienka dôkazu: Pre slová (grafy) s počtom uzlov menším resp. rovným $\max(rhs(p))^{|N|+1}$ platí triviálne. Ak je slovo resp. graf z jazyka väčší ako $\max(rhs(p))^{|N|+1}$, potom v jeho strome odvodenia sa nachádza



Obrázok 4.1: (a) p-strom odvodenia, (b) c-strom odvodenia

cesta na ktorej sa opakuje nejaký neterminál (maximálna hĺbka stromu bez opakovania neterminálu je $|N|$ a maximálny počet uzlov pre takýto strom odvodenia je $\max(rhs(p))^{|N|}$). Skrátením takejto cesty (stotožnením dvoch opakujúcich sa neterminálov (pravidiel), medzi ktorými sa na ceste neopakuje žiadny iný neterminál) vznikne slovo (graf), ktoré je menšie o menej ako $\max(rhs(p))^{|N|+1}$ terminálov. Skrátením vznikne tiež slovo z $L(G)$.

Poznámka 4.1.1. Nasledujúci dôkaz, pre svoju jednoduchšiu formuláciu, bude popisovať uzly stromu odvodenia neterminálmi namiesto pravidiel alebo kópií pravidiel. Definícia c-stromu odvodenia je akceptovateľná aj v podobe značenia neterminálnymi uzlami grafu.

Dôkaz. Ak G je C-edNCE gramatike s blokujúcimi hranami existuje k nej gramatika bez blokujúcich hrán podľa vety 2.4.2. Nech nová G je v CHNT, potom $\max(rhs(p))$ je 2. Uvažujme najdlhšie slovo v $L(G)$ také, že v jeho strome odvodenia sa na žiadnej ceste z koreňa do listu {stačí cesta z koreňa...} neopakujú dva rovnaké neterminály. Jednému stromu odvodenia v edNCE môže vo všeobecnosti prislúchať viac slov z $L(G)$. Avšak počet listov zostáva zachovaný.

Pre tento dôkaz použijeme nasledovnú pomocnú lemu.

Lema 4.1.1. Pre každý strom odvodenia v ktorom neexistuje cesta z koreňa do listu, ktorá by obsahovala dva rovnaké neterminály (pravidlové uzly) platí:

- jeho hĺbka je menšia nanajvýš rovná počtu neterminálov
- každé slovo prislúchajúce tomuto stromu odvodenia má maximálne $(\max(rhs(p)))^{|N|+1}$ terminálnych uzlov, kde $\max(rhs(p))$ maximálny počet uzlov na pravej strane pravidla a $|N|$ je počet neterminálov gramatiky G .

Dôkaz. Nech jeho hĺbka je väčšia ako $|N|$. Potom existuje cesta z koreňa do nejakého listu v hĺbke $|N| + k$, kde $k \geq 1$. Táto cesta určite obsahuje aspoň dva rovnaké neterminály. { dirichlet }

Každý neterminálny uzol je nahradený maximálne $\max(rhs(p))$, pretože počet uzlov na pravej strane každého pravidla je menší nanaajvýš rovný $\max(rhs(p))$. Pre strom hĺbkou $|N|$ platí:

- počet uzlov vo vetnej forme prislúchajúcej stromu odvodu s hĺbkou 1 je menší, nanaajvýš rovný $\max(rhs(p))$
- pre hĺbku $|N|$ je to $\max(rhs(p))^{|N|}$

Inak povedané $\max(rhs(p))$ predstavuje maximálne vetvenie v strome odvodu. Úplný strom s vetvením $\max(rhs(p))$ má $\max(rhs(p))^{|N|+1}$ listov.

□

Pokračujeme v dôkaze vety 4.1.1.

Nech w_j a w_i sú počtom uzlov susedné slová z G .

Bez ujmy na všeobecnosti: nech $w_j \geq w_i$. Uvažujme dva prípady:

- $|w_j| \leq \max(rhs(p))^{|N|+1}$:

Keďže $w_i \leq w_j \leq \max(rhs(p))^{|N|+1}$, potom $|w_i| + \max(rhs(p))^{|N|+1} \geq |w_j|$ a tvrdenie vety platí.

- $|w_j| > \max(rhs(p))^{|N|+1}$:

Sporom: Nech $|w_i| + \max(rhs(p))^{|N|+1} < |w_j|$. Podľa vyššie uvedenej lemy, obsahuje strom odvodu slova w_j podstrom hĺbkou $|N|+1$, v ktorom sú dva rovnaké neterminály na nejakej ceste z koreňa podstromu do listu. Bez ujmy na všeobecnosti označme α neterminál opakujúci sa v strome odvodu slova w_j . Nech je to najbližšia opakujúca sa dvojica neterminálov na ceste v strome (žiadna iná dvojica neterminálov sa medzi nimi neopakuje). Odstránením tejto cesty zo stromu resp. stotožnením rovnakých neterminálnych uzlov dostávame strom odvodu, ktorého výsledné slovo je menšie o d uzlov. Pričom $d < \max(rhs(p))^{|N|+1}$, podľa vyššie uvedenej lemy a z CHNT gramatiky vyplýva že, $d > 0$

(inak by sme možno museli zlučovať viac rovnakých dvojíc neterminálov, aby sme zmenšili vetnú formu). Spor s tým, že slová w_j a w_i sú susedné. Odstránením časti cesty medzi dvoma rovnakými neterminálmi vznikne tiež slovo z jazyka $L(G)$. V novom strome odvodenia sme totiž použili len pravidlá z gramatiky G , pričom nadväznosť pravidiel je zaručená nadväznosťou pravidiel predošlého stromu a tým, že sme zlúčili cestu medzi dvoma rovnakými neterminálmi. Tým sme vetu 4.1.1 dokázali.

□

Veta 4.1.2. Pre počty hrán v C-edNCE jazykoch podobné tvrdenie o medzere neplatí.

Dôkaz. Ako dôkaz si uvedieme jazyk kompletných grafov K_n . Medzera v počte hrán sa s rastúcim počtom uzlov zväčšuje. Teda sa nedá vo všeobecnosti ohraničiť konštantou závislou na gramatike.

- počet hrán $\binom{1}{2}$ pre K_1
- počet hrán $\binom{2}{2}$ pre K_2
- počet hrán $\binom{3}{2}$ pre K_3
- počet hrán $\binom{4}{2}$ pre K_4
- ...
- počet hrán $\binom{n}{2}$ pre K_n

□

Aký veľký je nárast počtu hrán? Pre hrany tvrdenie o medzere neplatí z toho dôvodu, že prepisovanie hrán a tým aj ich počet sú, na rozdiel od uzlov, závislé aj na množine spájacích inštrukcií, ktorá je vo všeobecnosti definovaná binárnou reláciou medzi uzlami. Teda množstvo vznikajúcich, resp. zanikajúcich hrán je dané nielen pravou stranou pravidla, ako to bolo pre počet uzlov, ale aj množstvom susedov prepisovaného uzla v materskom

grafe. Susedných uzlov môže byť neobmedzený počet. Z toho istého dôvodu, ak nedovolíme vo vetnej forme hyperhrany, tak nárast počtu hrán v slove nebude väčší, ako počet susedov v aktuálne prepisovanej vetnej forme. Inak povedané, nárast počtu bude maximálne lineárny vzhľadom na počet uzlov vo vetnej forme.

Ďalším kvantitatívnym parametrom grafov je stupeň uzlov. Ukážeme, že pre maximálny stupeň grafov platí veta o medzere.

Veta 4.1.3 (Veta o medzere pre maximálny stupeň). Nech jazyk $L(G) \in \text{C-edNCE}$, P nech je postupnosť všetkých slov z $L(G)$, usporiadaná podľa maximálneho stupňa v slovách. Potom pre každú dvojicu a, b , bezprostredne po sebe nasledujúcich slov z P platí: $|max(st_b) - max(st_a)| \leq max(st(rhs(p))) * ||\Sigma| - |\Delta|| + 1$, kde st_a a st_b sú maximálne stupne grafov a , b a $max(st(rhs(p)))$ je maximálny stupeň pravých strán pravidiel gramatiky G .

Dôkaz. V odvodení každého slova C-edNCE gramatikou sa maximálny stupeň vetnej formy v každom kroku výpočtu zvýši maximálne o konštantu závislú iba na parametroch gramatiky, presnejšie na pravej strane pravidla. Predpokladajme CHNT gramatiky G . Potom stupeň uzla sa zvyšuje v závislosti na stupni uzla pravej strany použitého pravidla a jeho pripojení na susedov materského grafu. Intuitívne, zvýšenie stupňa je maximálne o takú konštantu K , kde K je maximálny stupeň uzlov pravej strany všetkých pravidiel. Preto dôkaz tejto vety bude analogický a prebehne len s malými zmenami oproti dôkazu vety 4.1.1. Ohraničenie medzery v tomto prípade je dané ako $max(K) * (|N| + 1)$, kde $max(K)$ je hodnota maximálneho stupňa uzla zo všetkých pravých strán pravidiel. $\square$

Veta 4.1.4. Nech jazyk $L(G) \in \text{neblokujúcich edNCE}$, P nech je postupnosť všetkých slov z $L(G)$, usporiadaná podľa počtu uzlov v slovách. Potom pre každú dvojicu a, b bezprostredne po sebe nasledujúcich slov z P platí: $||V_b| - |V_a|| \leq max(rhs(p))^{||\Sigma| - |\Delta|| + 1}$, kde V_a a V_b sú množiny uzlov pre a, b a $max(rhs(p))$ je maximálny počet uzlov pravej strany pravidiel gramatiky G .

Dôkaz. Dôkaz tejto vety je založený na rovnakom princípe ako dôkaz rovnakého tvrdenia pre C-edNCE gramatiky (veta 4.1.1). Preto aj v tomto prípade uvedieme len zdôvodnenie použiteľnosti tejto konštrukcie dôkazu.

Na rozdiel od C-edNCE gramatík, kde každé odvodenie je ekvivalentné nejakému najľavejšiemu odvodeniu nejakej edNCE gramatiky, podľa vety 2.4.1 o $L\text{-edNCE} = C\text{-edNCE}$, môže vo všeobecnosti k jednému stromu odvodenia prislúchať viac slov z jazyka. Táto vlastnosť edNCE gramatík nerobí pri našich úvahách problém, lebo všetky slová sú rovnako dlhé a líšia sa len v počte hrán. Presnejšie, majú rovnaký počet uzlov, ktorý je rovný počtu listov v strome odvodenia a líšia sa len poradím použitých pravidiel, čiže vznikli rôznym prechádzaním tohoto stromu. Podobne ako v tvrdení pre C-edNCE gramatiky nám nevádi, že jedno slovo môže mať viac stromov odvodenia. Tento fakt nehral pri našich úvahách žiadnu úlohu, lebo gramatika bola v CHNT. CHNT tvar nám zaručoval, že sme nemuseli viacnásobne zlučovať opakujúce sa neterminály na to, aby sme zmenšili veľkosť vetnej formy v konštrukcii dôkazu. Pre edNCE vo všeobecnosti nemusí existovať CHNT, preto pri rovnakej konštrukcii dôkazu sa v niektorých prípadoch môže stať, že budeme musieť viacnásobne zlučovať opakujúce sa dvojice neterminálov na to, aby sme zmenšili veľkosť vetnej formy, resp. počet listov stromu odvodenia. Epsilon pravidlá sú len špeciálnym prípadom. Pri ich použití zanikajú niektoré vetvy stromu odvodenia, čiže strom odvodenia zmenšujú. Navyše, epsilon (vymazávacie) pravidlá sa dajú odstrániť aj v edNCE gramatikách, a to rovnakou konštrukciou ako pre C-edNCE gramatiky. $\square$

Veta 4.1.5. Pre blokujúce edNCE gramatiky tvrdenie o medzere medzi počtom uzlov susedných slov usporiadanej postupnosti jazyka generovaného gramatikou, neplatí.

Dôkaz. Blokujúce hrany tvoria niečo ako spôsob filtrovania terminálnych vetných foriem. Graf, ktorý obsahuje len terminálne uzly, ale obsahuje neterminálnu hranu nepatrí do jazyka generovaného gramatikou. Takáto hrana sa nazýva blokujúca, pretože vzhľadom na to, že sa nachádza medzi terminálnymi uzlami sa nedá odstrániť a ani prepísať. Takisto na ne možno pozeráť, ako na spôsob vynútenia nejakej postupnosti pravidiel, ktorá vedie k odvodeniu

slova patriaceho do jazyka generovaného gramatikou. Nasledujúci príklad, slúžiaci ako dôkaz tejto vety, využíva uvedené schopnosti blokujúcich hrán.

Na dôkaz tohto tvrdenia uvidíme jazyk, ktorý posluží ako kontra príklad k tvrdeniu o medzere medzi uzlami. Konkrétne je to jazyk K_n s $n = 2^d$ generovaný blokujúcou edNCE gramatikou. Pre tento jazyk neplatí veta 4.1.1, a aj preto nepatrí do triedy jazykov C-edNCE, resp. neblokujúcich edNCE jazykov.

Gramatika pre tento jazyk vyzerá nasledovne: $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$

$$\Sigma = \{X, Y, x\}, \Delta = \{x\},$$

$$\Gamma = \{\alpha, \beta, \gamma, \delta, \omega, \phi\},$$

$$\Omega = \{\delta\}, S = X$$

$$P = \{$$

$$X \rightarrow D = (X, \alpha, Y), C = \{(Y, \alpha/\beta, X), (Y, \alpha/\alpha, Y), *\}$$

$$X \rightarrow D = x, C = \{(Y, \alpha/\gamma, x), *\}$$

$$Y \rightarrow D = (Y, \alpha, Y), C = \{(X, \beta/\alpha, Y), (Y, \alpha/\alpha, Y), *\}$$

$$Y \rightarrow D = x, C = \{(Y, \alpha/\delta, x), (x, \gamma/\delta, x), *\}$$

* pre ostatné dvojice uzlov spájacie inštrukcie sú tvaru (ϕ/ω) , okrem finálnej hrany δ , ktorá sa len prepisuje. }

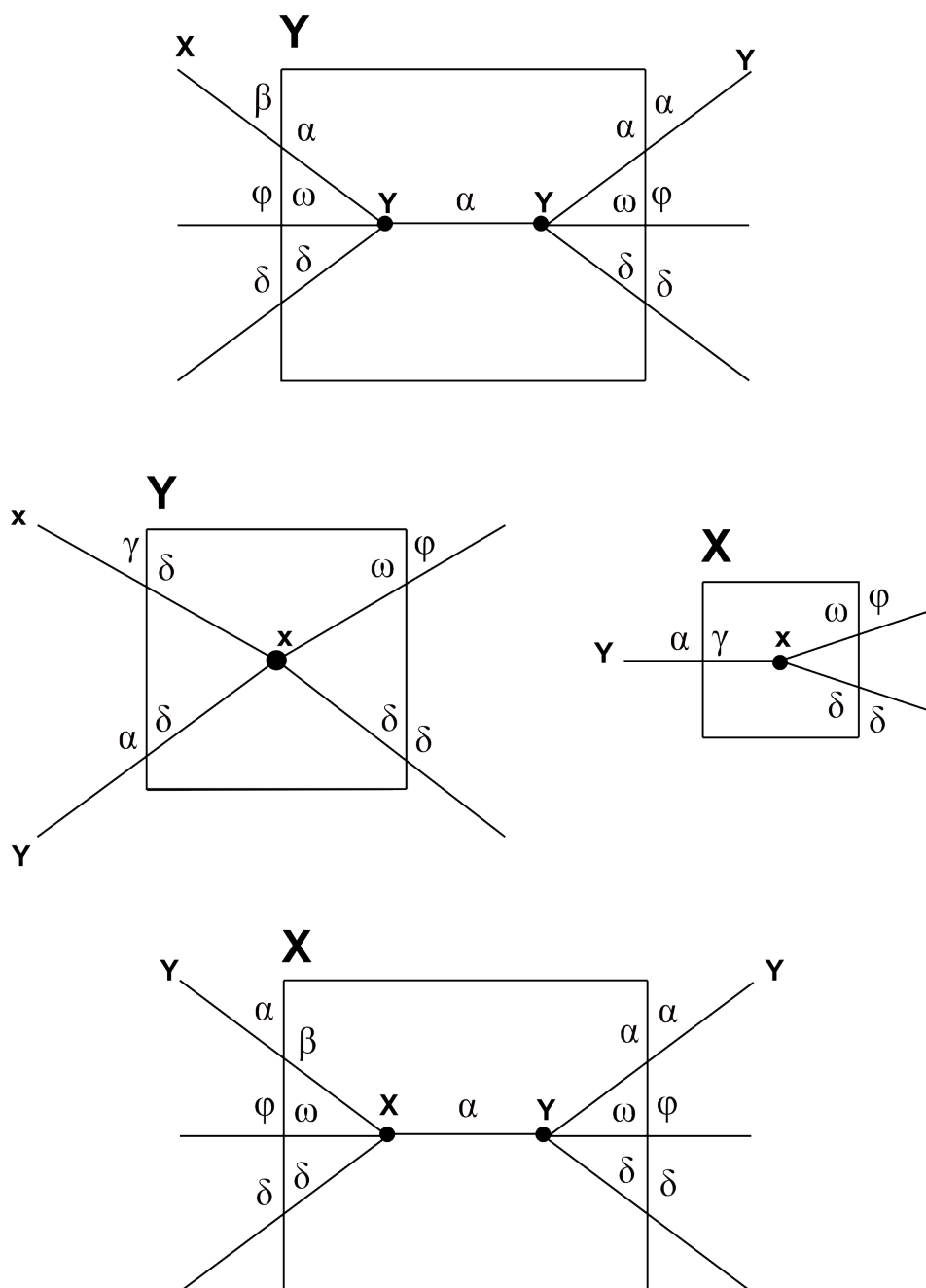
Vyššie popísaná gramatika pracuje nasledovným spôsobom. Pre jednoduchšie pochopenie neuvádzame podrobne všetky hrany medzi uzlami v popise generovania grafov gramatikou. Druhý a podstatnejší dôvod tohto kroku je ten, že gramatika generujúca kompletne grafy musí v každej vetnej forme svojho odvodenia generovať kompletne graf. Hrana, ktorá zanikne medzi dvoma existujúcimi uzlami sa nedá obnoviť bez nejakej kontextovej informácie navyše. Táto vlastnosť vyplýva z definície NCE prepisovacieho systému, presnejšie z jej definície množiny spájacích inštrukcií (neighbourhood controlled embedding).

Z počiatočného neterminálu S sa odvodí, buď graf s jedným uzlom a terminálnym symbolom x , ktorý patrí do jazyka, alebo graf s dvoma uzlami označenými neterminálmi X a Y . Blokujúce hrany v tejto gramatike slúžia na vynútenie postupnosti prepisov pomocou pravidiel nasledovne. Uzol X je niečo ako koordinátor rastu vetnej formy. „Koordinátor“, resp. uzol X je spojený hranou s každým uzlom Y (kompletný graf). Pri každom prepise

uzla X , resp. použitím pravidla, ktoré ho prepisuje na dva uzly, konkrétne X a Y , sa vytvára spojenia medzi uzlom X dcérskeho grafu a všetkými uzlami Y materského grafu takými hranami, ktoré pri použití iného pravidla ako toho, ktoré je očakávané (chceme, aby sa Y tiež prepísal na dva nové uzly Y a Y), zaručí vznik blokujúcej hrany. Blokujúca hrana, ako sme už raz spomenuli, sa nebude dať odstrániť a ani prepísať. Vlastnosť blokovania v gramatike sme zaručili vhodne zvolenými množinami inštrukcií pre jednotlivé pravidlá. Týmto spôsobom sme získali to, že sa najskôr prepíše uzol X , a až potom ostatné uzly Y , okrem uzla Y , ktorý vznikol pri prepise X . Rovnakým princípom zabezpečíme prepisovanie vetnej formy aj pri prepise uzla X na terminálny uzol x . Presnejšie, vždy sú zvolené vhodné pravidlá, resp. postupnosti pravidiel na prepisovanie jednotlivých uzlov vetnej formy. V našom prípade, buď sa uzol X prepíše na dva nové uzly X a Y a následne aj ostatné uzly Y na dva nové neterminálne uzly Y , alebo sa uzol X prepíše na terminálny uzol a potom takisto aj všetky ostatné uzly Y .

Pravidlá pre túto gramatiku sú zobrazené na obrázku 4.2. Štvorec symbolizuje uzol, ktorý je prepisovaný podľa pravidla. Všetko čo sa nachádza mimo jeho vnútra symbolizuje ľavú stranu pravidla. V našom prípade je to uzol, a všetky jeho hrany prepisujúce sa podľa množiny spájacích inštrukcií použitého pravidla. Vnútro štvorca zobrazuje pravú stranu pravidla aj s príslušnými prepismi, resp. preznačeniami hrán podľa množiny spájacích inštrukcií pravidla.

Popis k obrázku 4.2 gramatiky pre K_n , $n = 2^m$, $m \in \mathbb{N}$: ϕ je abstrakcia pre akúkoľvek hranu, ktorá nebola definovaná pred tým v obrázku pre pravidlá. Presnejšie, pre konkrétne pravidlo označené štvorčekom. Hrana ϕ signalizuje, že vetná forma sa prepísala v neočakávanom poradí použitia pravidiel, alebo bolo použité neočakávané pravidlo. Preto sa hrana ϕ prepisuje na hranu ω , ktorá je nefinálna a v našom prípade aj blokujúca, pomocou inštrukcie spojenia (ϕ/ω) . Hrana ω sa pomocou inštrukcie (ϕ/ω) , definovanej v každom pravidle, bude prepisovať len na samú seba. Hrana δ je finálna a je jediná. V skutočnosti sa táto hrana už neprepisuje, lebo je finálna. Kvôli názornosti a jednoduchosti popisu blokujúcich hrán pri generovaní vetnej formy je definovaný jej prepis inštrukciou (δ/δ) . $\square$



Obrázok 4.2: Pravidlá pre gramatiku

4.2 Zhrnutie

Pre **C-edNCE** gramatiky veta o medzere pre:

- uzly - platí
- hrany - neplatí
- stupeň uzla - platí

Pre neblokujúce **edNCE** gramatiky veta o medzere pre:

- uzly - platí
- hrany - neplatí
- stupeň uzla - platí

Pre **edNCE** gramatiky aj s blokujúcimi hranami veta o medzere pre:

- uzly - neplatí ($K_n, n = 2^a, a \geq 0$)
- hrany - neplatí
- stupeň - neplatí

Hypotéza:

Pre každý prepisovací systém, ktorý spĺňa nasledujúce vlastnosti:

- atomickosť
- lokálnosť
- a je bez podmienok na použitie prepisovacieho pravidla

platí veta o medzere analogicky ako pre neblokujúce edNCE a C-edNCE jazyky.

Jeden z problémov pri úvahách o platnosti tejto hypotézy je prienik ľavých strán pravidiel.

Platnosť predchádzajúcich výsledkov o medzere pre nejaký prepisovací systém sa dá vo všeobecnosti chápať ako nutná podmienka príslušnosti do

triedy jazykov definovaných týmto systémom. Avšak, vzhľadom na väčšiu všeobecnosť tvrdenia o medzere, v porovnaní s nutnými podmienkami príslušnosti do triedy jazykov nad reťazcami (pumpovacie lemy pre regulárne a bezkontextové jazyky), je toto tvrdenie len ich slabším zovšeobecnením. Pumpovacie lemy pre triedy jazykov nad reťazcami využívali napríklad rozdiel v schopnosti koordinácie prepisovania v rôznych častiach vetnej formy regulárnymi, bezkontextovými a kontextovými gramatikami pre dôkaz nepríslušnosti jazyka do danej triedy. Táto vlastnosť, čo sa týka uzlov, sa dá jednoducho zabezpečiť pomocou hrán v celej vetnej forme, alebo na hociktorom mieste vetnej formy generovanej grafovou gramatikou, ako bude neskôr ukázané a ilustrované na obrázkoch 5.1 a 5.2. Ďalším rozdielom medzi vetou o medzere a pumpovacou lemovou je, že pumpovacia lema, na rozdiel od vety 4.1.1, popisuje štruktúru vetnej formy jazyka. Pumpovacia lema totiž popisuje opakujúce sa časti vetnej formy ako akési vzory.

Poznámka 4.2.1. Klasická pumpovacia lema je známa pre NLC grafové jazyky v tvare, kde sa nepovoľuje nulové pumpovanie. Pre B-NLC platí aj s nulovým pumpovaním. V edNCE prepisovacom systéme je známa pre HR, čiže $B_{nd} - edNCE$ triedu jazykov.

4.2.1 Zhrnutie vlastností

Vlastnosti C-edNCE jazykov

C-edNCE gramatiky generujú jazyky, ktoré sú pravidelné, čo sa týka ich rozdielu v počte uzlov susedných slov usporiadanej postupnosti jazyka (podľa vety 4.1.1). A takisto sú do istej miery pravidelné aj v počte hrán, resp. ich rozložení (podľa vety 2.5.3). Všetky nekonečné podpostupnosti slov z C-edNCE jazyka, ktoré sa dajú rozumne popísať (sú MSO definovateľné), majú podobné vlastnosti (z vety 4.1.1) ako C-edNCE jazyk do ktorého patria. Skladajú sa len z takých rozumne definovateľných postupností grafov, ktoré sú tiež bezkontextové. C-edNCE gramatiky nedokážu generovať jazyky, ktoré obsahujú veľa grafov s rovnakým počtom uzlov a líšiacimi sa len rozložením alebo len počtom hrán medzi svojimi uzlami (zhrnutie z viet 2.5.3, 4.1.1, 5.1.1

a lemy 5.1.2). Pre jazyk generovaný C-edNCE gramatikou platia nasledovné vlastnosti:

1. Relatívne malá hustota grafov s rovnakým počtom uzlov, maximálne $\binom{n+k-1}{k-1}$, kde k je konštanta závislá na parametroch gramatiky. Táto vlastnosť je v kontraste s bezkontextovými jazykmi nad reťazcami.
2. Relatívne veľká hustota grafov s rôznym počtom uzlov (veta 4.1.1).

Vlastnosti neblokujúcich edNCE jazykov

Neblokujúce edNCE gramatiky, podobne ako C-edNCE, nedokážu generovať grafy s veľkými rozdielmi v počte uzlov. Na rozdiel od C-edNCE sa však nimi generované jazyky môžu skladať z nebezkontextových podpostupností grafov (nekonfluentných). Dôsledok vety 2.5.3, príkladom jazyka s touto vlastnosťou sú všetky grafy nad danou abecedou uzlov a hrán. Tento jazyk obsahuje okrem iných napríklad postupnosť (jazyk) štvorcových mriežok, ktorá je MSO definovateľná, ale nie je bezkontextová. Teda podľa vety 2.5.3, resp. uzavretosti C-edNCE jazykov na prienik s MSO definovateľnými grafovými jazykmi, jazyk všetkých grafov nad danými abecedami nepatrí do C-edNCE. Neblokujúce edNCE gramatiky dokážu generovať jazyky s veľkou variabilitou hrán medzi uzlami. Inak povedané, vedia generovať veľké množstvo grafov, ktoré sa líšia len v počte a rozložení hrán. Takisto generujú aj také jazyky, ktoré sú tvorené relatívne veľkým množstvom grafov s rovnakým počtom uzlov. Príkladom je opäť jazyk všetkých grafov nad danou abecedou uzlov a hrán.

Vlastnosti blokujúcich edNCE jazykov

Blokujúce edNCE jazyky sú oproti neblokujúcim edNCE jazykom rozšírené o jazyky s exponenciálnymi medzerami medzi susednými slovami, čo sa týka počtu hrán.

Problémy pri skúmaní štruktúry grafových jazykov

Presnejšiemu popisu štruktúry grafov generovaných jednotlivými hierarchiami grafových gramatík brania nasledujúce vlastnosti prepisovacieho systému. Spôsob rozloženia hrán v grafoch, resp. spôsob prepisovania používajúci množinu spájacích inštrukcií je ťažko popísať kvôli jeho dynamickej povahe vytvárania hrán medzi jednotlivými uzlami grafu v rôznych častiach vetnej formy. Presnejšie, nie v každom kroku odvodenia sa využije každá inštrukcia spojenia použitého pravidla. Preto pri použití toho istého pravidla v rôznych častiach vetnej formy môže výsledok nahrádzania môže viesť k vytvoreniu inej, pravidlom vytvárajúcej časti grafu. Táto sa síce nebude líšiť v počte uzlov, môže sa však líšiť v počte, alebo rozložení hrán, ktoré ju spájajú so zvyškom grafu.

Kapitola 5

Usmerňovanie prepisovania

V klasickej teórii formálnych jazykov sa skúmali rôzne spôsoby „riadenia odvodenia“ v gramatikách a ukázalo sa, že riadenie odvodenia môže zvýšiť generatívnu silu gramatík. V tejto kapitole uvidíme naše výsledky a úvahy o tom, ako sa dajú odvodenia v grafových gramatikách „usmerňovať“. Zameriame sa na dva spôsoby „vynucovania“ poradia použitých pravidiel. Jedným sú blokujúce hrany, tým druhým riadiace jazyky (analogické niektorým odvodeniám v gramatikách pre reťazce).

5.1 Blokujúce hrany a regulárne gramatiky riadenia odvodenia v C-edNCE gramatikách

Vo všeobecnosti majú grafové gramatiky s riadiacou gramatikou väčšiu resp. aspoň takú istú generatívnu silu ako grafové gramatiky s blokujúcimi hranami. Otázkou však je, že aká silná riadiaca gramatika bude stačiť. V ďalších častiach tejto diplomovej práce sa budeme zaoberať porovnaním regulárnych gramatík riadenia odvodenia a blokujúcich hrán nad prepisovacími systémami neblokujúce edNCE a C-edNCE. Podľa definície konfluencie gramatiky 2.2.4, v odvodení C-edNCE gramatiky nezáleží na poradí použitých pravidiel. Pri ľubovoľnej permutácii postupnosti pravidiel, umožnenej nadväznosťou pravidiel vo vetnej forme, resp. gramatike musí vznikať to isté slovo, alebo tá istá vetná forma. Z tohto dôvodu môžeme svoje úvahy o odvodeniach v

C-edNCE gramatikách zúžiť z postupností pravidiel na multimnožiny pravidiel. Tento fakt sme už využili v dôkaze vety 4.1.1 a neskôr ešte využijeme pri porovnaní blokujúcich hrán a regulárnych gramatík riadenia odvodenia v C-edNCE prepisovacom systéme, konkrétne v leme 5.1.2 a vo vete 5.1.1.

- Pre C-edNCE gramatiky platí obmedzenie na veľkosť rozdielu v počte uzlov medzi susednými slovami, ktoré generujú. Ako sme už spomenuli, tento rozdiel je ohraničený konštantou závislou na gramatike (veta 4.1.1). Preto hovoríme o relatívne veľkej hustote grafov s rôznym počtom uzlov v jazykoch C-edNCE. To isté obmedzenie pochopiteľne platí aj pre postupnosti a multimnožiny pravidiel C-edNCE gramatiky.
- Druhým obmedzením, ktoré sme spomenuli ale neupresnili, je relatívne malá hustota grafov s rovnakým počtom uzlov. Uvažujme C-edNCE gramatiku v CHNT a bez Λ -pravidiel, s k pravidlami. Potom počet rôznych multimnožín pravidiel s rovnakou mohutnosťou je ohraničený zhora: $\binom{n+k-1}{k-1}$, čo je počet n prvkových multimnožín skladajúcich sa z k rôznych prvkov. Pre slová generované gramatikou v CHNT a bez Λ -pravidiel toto obmedzenie na počet multimnožín udáva aj obmedzenie na počet grafov resp. slov s rovnakým počtom uzlov, lebo dĺžka odvodenia slova veľkosti n uzlov je maximálne $2n$. Počet rôznych v zmysle neizomorfných grafov s n uzlami bez značenia hrán a uzlov je 2^{n-1} , čo je asymptoticky viac ako $\binom{n+k-1}{k-1}$. Ako sme už spomenuli, C-edNCE gramatika nedokáže generovať veľa grafov s rovnakým počtom uzlov.

Pre porovnanie možnosti zvýšenia generatívnej sily edNCE gramatík regulárnymi gramatikami riadenia odvodenia a blokujúcimi hranami, je pre nás kľúčové zistiť, akým spôsobom môže dané rozšírenie prepisovacieho systému zvýšiť jeho generatívnu silu. Generatívna sila akéhokoľvek prepisovacieho systému spočíva v tom, aké množstvo jazykov, nad danou doménou dokáže daný systém generovať. Voľne napísané, doména generatívneho systému je triedy všetkých jazykov, pre ktoré má potenciálnu možnosť ich generovania. Za istých okolností sa dá na tento fakt pozeráť aj ako na to, koľko vetných foriem dokáže daný prepisovací systém vynechať, čiže odfiltrovať z jazyka.

Pre porovnanie blokujúcich hrán a regulárnych gramatík riadenia odvodenia nad C-edNCE prepisovacím systémom je v našom prípade dôležitejší druhý pohľad na generatívnu silu modelu. Snažíme sa zistiť, koľko terminálnych vetných foriem sme pomocou blokujúcich hrán, resp. regulárnej gramatiky riadenia odvodenia schopní z jazyka vynechať, alebo aké zložité budú postupnosti a množiny slov generované gramatikou rozšírenou o blokujúce hrany alebo riadiacu gramatiku. Prvý pohľad na generatívnu silu modelu je daný jeho definíciou a tvorí akýsi náhľad na doménu generovaných jazykov. Druhý pohľad slúži, okrem iného, aj k vytvoreniu obrazu o hierarchii medzi triedami jazykov generovaných jednotlivými prepisovacími systémami. Stručne povedané k odlíšeniu jednotlivých tried jazykov. V oboch prípadoch sú to v podstate pohľady na prepisovací systém a jeho obmedzenia, ktoré ho nútia generovať jazyky s určitými vlastnosťami a štruktúrou. Napríklad edNCE je prepisovací systém a C-edNCE je prepisovací systém s vlastnosťou (obmedzením) konfluencie.

5.1.1 Skúmané vlastnosti

Kvôli náročnosti priameho porovnania blokujúcich hrán a riadiacich gramatík nad grafovými gramatikami, použijeme pre odlíšenie generatívnej sily jedného prepisovacieho systému od druhého, nasledujúce vlastnosti pri obmedzeniach resp. rozšíreniach prepisovacích modelov.

- Schopnosť koordinácie prepisovania, čo znamená kontrolu nad počtom uzlov v rôznych častiach vetnej formy. Táto vlastnosť je pre nás nezaujímavá, pretože sa dá v jazykoch generovaných grafovými gramatikami, vďaka hranám, jednoducho docieľiť. Pre jazyky nad reťazcami sme museli na zabezpečenie tejto vlastnosti zaviesť, resp. ponechať nejakú kontextovú informáciu v prepisovacom systéme.
- Miera zvýšenia hustoty grafov s rovnakým počtom uzlov, alebo schopnosť generovať viac grafov s rovnakým počtom uzlov.
- Miera zníženia hustoty grafov s rôznym počtom uzlov, alebo porušenie vety 4.1.1.

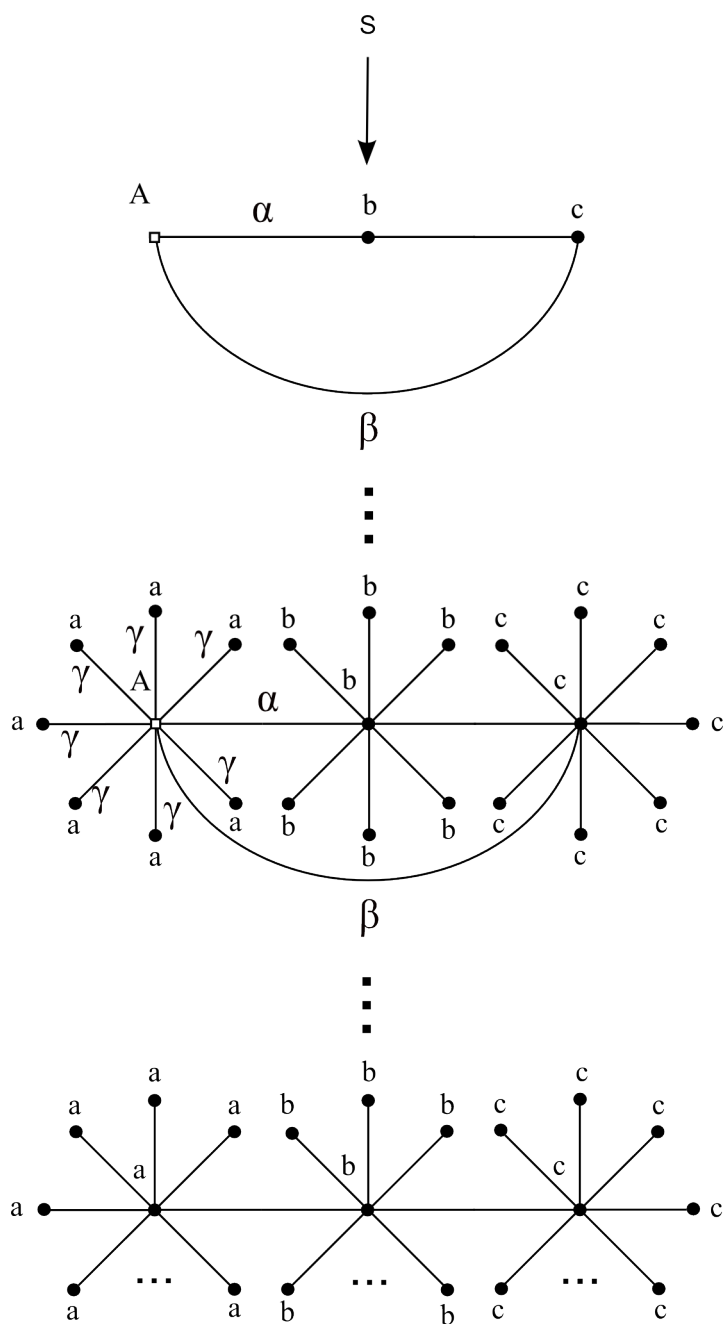
Uvedené vlastnosti prepisovacieho systému vypovedajú o jeho generatívnej sile.

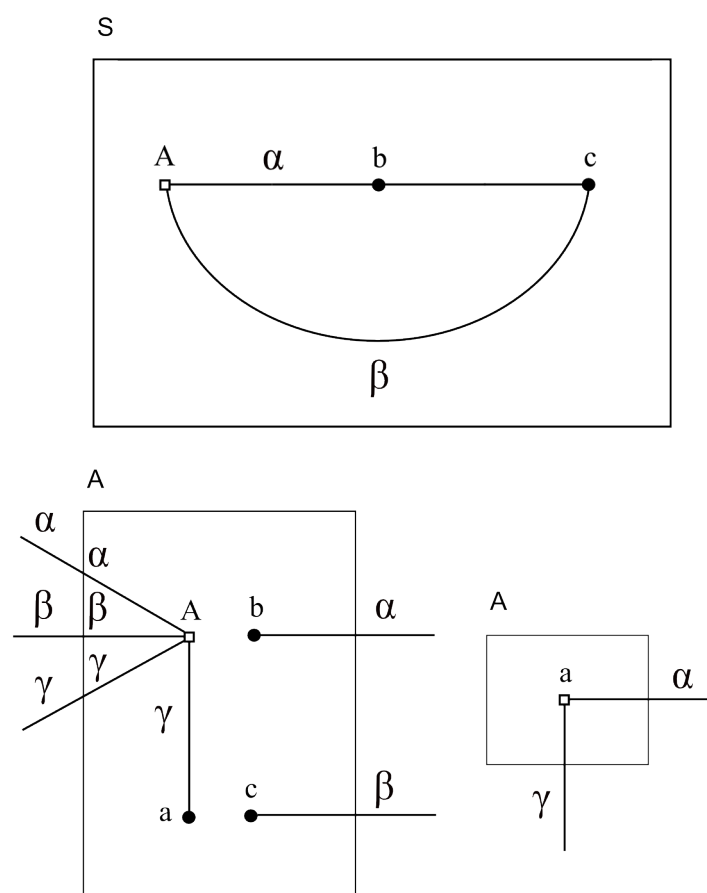
Schopnosť koordinácie

Koordinácia, ktorú dokážu zabezpečiť regulárne gramatiky riadenia odvodenia, priniesla väčšiu generatívnu silu pre regulárne gramatiky aj pre bezkontextové jazyky nad reťazcami. V grafových gramatikách edNCE prepisovacieho systému tento spôsob koordinácie žiadnu generatívnu silu nepridá. Koordinácia, v zmysle generovania rovnakých počtov uzlov s danými značkami, resp. rovnakých vetných foriem v rámci nejakých častí slov patriacich do jazyka (napríklad ww , www , $a^n b^n c^n$, ...) je schopný samotný prepisovací systém edNCE a dokonca aj C-edNCE. Táto koordinácia sa dá simulovať pomocou hrán a ich značenia medzi význačnými uzlami. Hrany môžu byť použité na označenie týchto význačných miest, na ktorých sa má časť vetnej formy synchronizovať s inými a ich značenie slúži na konkrétnu simuláciu koordinácie pri prepise pravidlom. Príklady koordinácie sú napríklad vynútenie rovnakých častí vetnej formy na určitých miestach alebo rovnaký počet určitých uzlov a podobne. Ako príklad použitia hrany na tento účel poslúži nasledujúci jazyk L , ktorý synchronizuje počty uzlov s rôznymi značkami. Konkrétne, L je jazyk s rovnakými počtami ulov rôznych značiek. Vetné formy vytvorené gramatikou pre tento jazyk sú popísané na obrázku 5.1.

Podľa vetných foriem zobrazených na obrázku nie je ťažké napísať jednoduchú C-edNCE gramatiku generujúcu tento jazyk. Mohla by vyzeráť napríklad ako nasledujúca z obrázka 5.2.

Záver: Koordinácia odvodenia vo vetnej forme je z hľadiska porovnania generatívnej sily jednotlivých modelov grafových gramatík a ich jazykov zväčša nepodstatnou vlastnosťou. Presnejšie je to tak u tried jazykov vytváraných prepisovacími modelmi, ktoré obsahujú značenie hrán, ich preznačovanie a inštrukcie pre spojenie pravidlom vytvoreného grafu so zvyškom vetnej formy pomocou hrán (eNCE, eNLC). Hrany a ich značenie nám umožnilo koordinovať vetnú formu aj v spomenutom prípade koordinácie eNCE gramatikou.


 Obrázok 5.1: Vetné formy jazyka L



Obrázok 5.2: Gramatika pre jazyk L

5.1.2 Porovnanie vlastností regulárnych gramatík riadenia odvodenia a blokujúcich hrán nad C-edNCE prepisovacím systémom:

Blokujúce hrany podľa vety 2.4.2 nepridajú žiadnu generatívnu silu danému prepisovaciemu systému a takisto nespôsobia žiadne obmedzenie jeho generatívnej sily. Presnejšie, ak zakážeme C-edNCE gramatike vznik blokujúcej hrany, čo znamená, že v odvodení nemôže existovať neterminálnu hrana medzi dvoma terminálnymi uzlami, potom generatívnu silu neobmedzíme. Otázkou zostáva, akú generatívnu silu, ak vôbec, môžu pridať regulárne gramatiky riadenia odvodenia.

Lema 5.1.1. Pre postupnosti pravidiel, generovaných riadením odvodenia pomocou regulárnej gramatiky, platí veta 4.1.1.

Dôkaz. Z pumpovacej lemy vieme, že existuje nejaká konštanta, ktorá ohraňuje rozdiel najbližších (podľa počtu terminálov uzlov) slov patriacich do regulárneho jazyka. Keďže najmenší rozdiel medzi postupnosťami vedúcimi k odvodeniu slova je ohraničený konštantou a vzhľadom na to, že je ohraničená aj veľkosť pravej strany pravidla, preto je ohraničený aj rozdiel medzi dvoma najbližšími slovami v jazyku grafovej gramatiky, čo sa týka počtu uzlov. Teda pre postupnosti pravidiel, ktoré sú generované regulárnym riadením odvodenia, platí veta 4.1.1. $\square$

Lema 5.1.2. Maximálny počet multimnožín pravidiel (slov) s daným počtom pravidiel (uzlov) je zhora ohraničený $\binom{n+k-1}{k-1}$.

Dôkaz. Maximálny počet rôznych pravidiel použitých v odvodení je k alebo je ohraničený počtom pravidiel gramatiky, ktorých je n . Maximálny počet pravidiel v C-edNCE gramatike, ktorá je v CHNT je daný dĺžkou odvádzaného slova. Hľadáme počet všetkých n prvkových multimnožín, ktoré majú k rôznych prvkov. Nie je ťažké nahliadnuť, že ich počet je $\binom{n+k-1}{k-1}$. $\square$

Veta 5.1.1. Regulárne gramatiky riadenia odvodenia nepomôžu C-edNCE resp. neblokujúcim edNCE gramatikám generovať jazyky, ktoré nespĺňajú

vetu 4.1.1. (Regulárne riadenie odvodenia neznižuje hustotu grafov s rôznym počtom uzlov v jazykoch).

Dôkaz. Podľa pumpovacej lemy, regulárne gramatiky zväčšujú veľkosť svojich slov (postupností pravidiel) maximálne o konštantu K_1 závislú na gramatike. Pumpovacia lema pumpuje časti slov o nejakú konštantu, pričom počet častí slov, ktoré sa dajú pumpovať je konečný až na ich nekonečné opakovania.

C-edNCE gramatiky zväčšujú veľkosť svojich slov a teda aj postupností pravidiel tiež maximálne o konštantu K_2 (veta 4.1.1).

Pre konkrétne konštanty $k_1 \leq K_1$, resp. $k_2 \leq K_2$ rastu postupnosti pravidiel regulárnej gramatiky, resp. C-edNCE gramatiky môžu nastať dve možnosti pre postupnosti pravidiel generovaných uvedenými gramatikami (v prípade C-edNCE môžeme uvažovať multimnožiny pravidiel).

1. majú spoločný prienik: Majú zhodné tie postupnosti pravidiel, ktoré sú násobkami jednotlivých konštánt (rastu postupnosti pravidiel v regulárnej gramatike a rastu multimnožiny pravidiel v C-edNCE gramatike). Pumpovanie časti vetnej formy v regulárnej gramatike a opakovanie cesty v strome odvodenia C-edNCE gramatiky.
2. nemajú spoločný prienik: Ak sú postupnosti rovnako dlhé, čo určite nastane v násobku konštánt rastu jednotlivých postupností, tak vzhľadom na to, že C-edNCE nedefinuje žiadne obmedzenia pre použitie pravidiel, musia sa tieto postupnosti pravidiel líšiť v nejakom konkrétnom pravidle postupnosti resp. nepoužiteľnosťou multimnožiny pravidiel definovaných regulárnou riadiacou gramatikou. V tom prípade môžeme vynechať všetky takéto postupnosti a ich rozšírenia o násobky konštánt.

Počet rôznych častí vetných foriem, ktoré sa dajú pumpovať je konečný, až na ich nekonečné opakovania. Podobne je to pre C-edNCE.

Počet najmenších spoločných násobkov zložených z jednotlivých konštánt regulárnej gramatiky a C-edNCE gramatiky je tiež konečný počet (simulovanie rôznych rozšírení postupností resp. generovanie všetkých postupností). Ostatné sú len ich zložením. Preto spoločným prienikom postupností

pravidiel regulárnej gramatiky a C-edNCE gramatiky sa filtruje maximálne konštantne veľa postupností pravidiel.

Iný dôkaz

Platnosť tejto vety sa dá ukázať aj z uzavretosti bezkontextových reťazcových jazykov na prienik s regulárnymi jazykmi. Myšlienka je založená na použití viet 2.4.3, uzavretosti bezkontextových reťazcových jazykov na prienik s regulárnymi jazykmi, vety 2.4.5 a poznámka 2.4.1 pre edNCE, vďaka ktorej môžeme použiť rovnakú úvahu, ako pre uzly v slovách grafového jazyka, aj pre postupnosti pravidiel (veta 4.1.1).

□

Zovšeobecnenie pre neblokujúce edNCE jazyky: Rovnaké úvahy ako vo vete 5.1.1 platia aj pre edNCE. Opakovanie sekvencií pravidiel aj v odvodení edNCE gramatiky vedie k tým istým úvahám, lebo nám nezáleží na tom ako bude vyzeráť výsledné slovo. Ak sa zhoduje postupnosť pravidiel v edNCE gramatike s nejakou z jeho regulárnej riadiacej gramatiky a zároveň sa zhoduje aj rozšírenie postupnosti edNCE o opakovanie sekvencií pravidiel v strome odvodenia s nejakou postupnosťou pravidiel z reg. riadiacej gramatiky, ktorá vznikla rozšírením pôvodnej postupnosti, potom je počet vynechaných postupností ohraničený konštantou, ktorá vznikne napríklad spoločným násobkom konštánt rozširovania pre edNCE a pumpovania pre regulárnu gramatiku riadenia odvodenia. Ak sa teda zhodujú nejaké dve rozšírenia postupností určitej dĺžky danej pumpovacou lemov pre regulárne gramatiky, potom sa budú zo zloženia konštánt z pumpovacej lemy pre regulárne jazyky a z vety 4.1.1 pre edNCE vynechávať len konštantné počty postupností edNCE. V opačnom prípade sa vynechajú všetky. Počet rôznych častí vetnej formy pre pumpovanie v regulárnej gramatike je tiež obmedzený. Z tohto všetkého dostávame, konštantné rozdiely v počte uzlov medzi susednými slovami jazyka.

Rozdiely medzi slovami môžu vďaka regulárnym gramatikám riadenia odvodenia narásť najviac o konštantu. Vyplýva to z konštantného rastu postupností riadiacej gramatiky (pumpovacia lema) a takisto z konštantného rastu postupností pravidiel odvodenia edNCE gramatiky (veta 4.1.1). Ak od

určitej konštanty závislej na parametroch regulárnej gramatiky riadenia, nastala zhoda s postupnosťami pravidiel edNCE, potom vďaka ich konštantnému rastu, vynechajú maximálne konštantný počet postupností medzi dvoma postupnosťami povolených gramatikou riadenia. Pumpovanie odvodenia regulárnej gramatiky riadenia a rozširovanie postupností edNCE sú navzájom rovnaké v každom vhodnom násobku svojho rozšírenia, alebo nie sú rovnaké pre žiadny násobok rozšírenia resp. pumpovania (od určitej konštanty závislej na parametroch gramatík).

Regulárne gramatiky riadenia odvodenia netvoria žiadne obmedzenie pre C-edNCE prepisovací systém (jazyk všetkých počtov rôznych znakov nad danou abecedou $a^*b^* \dots$).

Ukázali sme, že týmto smerom regulárne gramatiky riadenia odvodenia nerozširujú generatívnu silu C-edNCE prepisovacieho systému a zároveň, že regulárne gramatiky riadenia odvodenia netvoria v tomto smere žiadne obmedzenie pre C-edNCE gramatiky.

Veta 5.1.2. Riadenie odvodenia pomocou regulárnych gramatík neumožní C-edNCE gramatikám generovať jazyky, ktoré majú väčšiu hustotu grafov s rovnakým počtom uzlov, resp. negenerujú viac grafov s rovnakým počtom uzlov.

Dôkaz. Intuitívne zdôvodnenie tohto tvrdenia znie nasledovne. Regulárne gramatiky riadenia odvodenia tvoria spôsob filtrovania vetných foriem, a preto ich hlavnou funkciou je odstraňovať niektoré slová z jazyka generovaného grafovou gramatikou. Najviac slov v jazyku ostane, ak dovoľíme každé odvodenie, čiže pre tento účel nemusíme mať žiadnu gramatiku riadenia. Maximálny počet odvodení s rovnakou dĺžkou je asymptoticky $\binom{n+k-1}{k-1}$ (z CHNT, resp. z odstránenia Λ -pravidiel platí aj pre slová jazyka), pričom všetkých grafov nad n uzlami je asymptoticky zdola ohraničený $2^n - 1$. $\square$

5.1.3 Zhrnutie

Charakterizácia C-edNCE jazykov:

Ak je trieda C-edNCE jazykov charakterizovaná ako málo (v zmysle rozumnej pomaly rastúcej funkcie) slov rovnakej dĺžky s malými (konštantnými) medzerami medzi slovami, potom predchádzajúce úvahy o regulárnych gramatikách riadenia odvodenia a blokujúcich hranách vedú k ich ekvivalencii nad C-edNCE prepisovacím systémom.

Problémy porovnania:

Jeden z najpodstatnejších problémov generatívneho prístupu skúmania grafových jazykov je, ako popísať gramatikou štruktúru generovaných grafov. Generatívny prístup skúmania reťazcových jazykov, na rozdiel od grafových jazykov, nepotreboval riešiť až tak veľké problémy spájajúce sa s priestorovou štruktúrou slov v jazykoch nad reťazcami, kvôli ich jednoduchej priestorovej štruktúre v porovnaní s grafmi.

5.2 Blokujúce hrany a regulárne gramatiky riadenia odvodenia v edNCE gramatikách

Pri svojich úvahách o edNCE gramatikách nemôžeme využiť multimnožiny, lebo zámena poradia pravidiel v odvodení môže viesť k rôznym slovám z jazyka ako bolo uvedené v príklade 2.1.1. edNCE gramatiky s blokujúcimi hranami dokážu generovať grafové jazyky s exponenciálnymi medzerami medzi susednými slovami.

5.2.1 Porovnanie vlastností regulárnych gramatík riadenia odvodenia a blokujúcich hrán nad edNCE prepisovacím systémom

V tejto časti uvedieme niekoľko našich pozorovaní a úvah o schopnostiach blokujúcich hrán resp. regulárneho riadenia zvýšiť generatívnu silu grafových

gramatík.

Odstránenie blokujúcich hrán z edNCE prepisovacieho systému zníži jeho generatívnu silu. Blokujúce hrany danému prepisovaciemu systému umožňujú generovať napríklad jazyky so slovami dĺžky 2^n .

Blokujúca hrana umožňuje prepisovaciemu systému využívať malú kontextovú informáciu o okolí prepisovaného uzla. Ide v podstate o vynútenie pravidla na prepis určitého uzla, tak aby použité pravidlo svojou množinou spájacích inštrukcií nespôsobilo preznačenie hrany prislúchajúcej prepisovanému uzlu na blokujúcu. Znamená to, že blokujúca hrana môže byť využitá na prenos informácie medzi uzlami, ktoré sú spojené hranou a tak donútiť gramatiku aby prepisovala konkrétne miesta vetnej formy na základe postupností pravidiel, ktoré nevedú k prepisu hrany medzi danými uzlami na blokujúcu. Príkladom vynútenia takéhoto prepisu je jazyk pozostávajúci z kompletných grafov o 2^n uzloch (obr. 4.2).

Tieto hrany môžu vynucovať resp. povoľovať určité pravidlá a iné zas naopak zakazovať resp. pri ich použití vznikne blokujúca hrana a teda výsledná terminálna vetná forma bude obsahovať neterminálnu hranu, kvôli ktorej nebude patriť do jazyka generovaného danou gramatikou.

Všade, kde vznikne určitý uzol a hrana vieme vďaka blokujúcim hranám vynútiť použitie určitej postupnosti pravidiel na susedné uzly resp. daný uzol. Teda postupnosti pravidiel vynútené blokujúcimi hranami sú definované pre konkrétne miesta vo vetnej forme.

Regulárna gramatika riadenia odvodenia slúži na vynútenie určitých vopred definovaných postupností pravidiel pomocou riadiacej gramatiky. Postupnosti pravidiel definované regulárnou gramatikou riadenia vedú k odvodeniu slov patriacich do jazyka. Ostatné postupnosti pravidiel definované neblokujúcou edNCE gramatikou sú neplatné.

Postupnosť pravidiel v regulárnom riadení je definovaná vopred a preto, ak by sme chceli definovať pomocou riadiacej gramatiky postupnosť pravidiel pre konkrétne miesto vo vetnej forme, museli by sme zaručiť vznik resp. existenciu vhodného miesta (vznik uzla resp. hrany) pre použitie pravidla alebo postupnosti pravidiel vo vetnej formy. Z dôvodu nedeterminizmu pri použití pravidiel neblokujúcou edNCE gramatikou musíme takisto zaručiť

aj akúsi jednoznačnosť tohto miesta vo vetnej forme, ktorá by zaručovala použitie pravidiel postupnosti na konkrétnom resp. určenom mieste. Inak povedané, ak chceme simulovať správanie blokujúcich hrán musíme vedieť postupnosť pravidiel použiť jednoznačne a nie viacerými spôsobmi podľa prepisu rôznych miest vetnej formy.

Porovnanie na základe viet 5.1.2, 5.1.1 a koordinovania odvodenia grafovou gramatikou, ako v prípade C-edNCE, platí aj pre neblokujúce edNCE. Navyše však blokujúce hrany pridávajú generatívnu silu, ktorej podľa vety 5.1.1 regulárne gramatiky riadenia nie sú schopné. Podľa vety 4.1.5 blokujúce edNCE gramatiky sú schopné generovať grafy s exponenciálnymi rozdielmi v počte uzlov medzi susednými slovami.

5.3 Zhrnutie

V tejto kapitole sme ukázali niektoré schopnosti blokujúcich hrán a regulárnych gramatík pre riadenia odvodenia v C-edNCE a edNCE systéme prepisovania. Konkrétne sme ukázali, že regulárne gramatiky riadenia odvodenia nepomôžu C-edNCE a ani neblokujúcim edNCE gramatikám generovať exponenciálne rozdiely v počte uzlov susedných slov. Tiež sme uviedli, že vlastnosť koordinácie vetnej formy je, na rozdiel od reťazcov, v grafových jazykoch jednoducho simulovateľná alebo vynútiteľná pomocou prepisovania hrán jednotlivými pravidlami, preto neprináša očakávané výsledky pri odlišovaní generatívnej sily uvedených modelov prepisovania grafov. Vlastnosť hustoty grafov s rovnakým počtom uzlov a hustoty grafov s rôznymi počtami uzlov zostáva aj pod vplyvom riadenia odvodenia pomocou regulárnych gramatík nezmenená. Blokujúce hrany naopak dokážu ovplyvňovať hustotu grafov s rôznym počtom uzlov. Jednoznačné porovnanie blokujúcich hrán a gramatík pre riadenia odvodenia sa síce nepodarilo ukázať ale ukázali sme, že nad edNCE prepisovacím systémom má riadenie odvodenia pomocou regulárnej gramatiky len neporovnateľnú alebo slabšiu generatívnu silu ako blokujúce hrany. Nad C-edNCE v prípade platnosti uvedenej hypotézy sme ukázali rovnosť resp. nezávislosť C-edNCE na oboch rozšíreniach.

5.3.1 Otvorené otázky k výsledkom kapitoly

- Dá sa nejakým spôsobom potvrdiť, alebo vyvrátiť myšlienka simulácie postupností blokujúcich hrán a porovnanie prepisu konkrétneho miesta postupnosťou pravidiel vynútených blokujúcou hranou s globálnym prepisom postupnosťou pravidiel riadiacej gramatiky? Niektoré nepodstatné časti prepisov v globálnej sa dajú vynechať resp. sú zbytočné z dôvodu existencie viacerých miest pre použitie pravidla.
- Vzhľadom na to že, globálne postupnosti pravidiel definované regulárnou gramatikou riadenia nemajú vopred určené konkrétne miesto použitia pravidiel, preto môžu obsahovať podpostupnosti pravidiel, ktoré sa vzhľadom na vetnú formu prepisujú nedeterministicky. Sú tieto podpostupnosti z pohľadu regulárneho riadenia zbytočné? Myšlienka simulácie platí, len ak vieme zaručiť, resp. kontrolovať množstvo a typ uzlov resp. hrán prepisovaných vo vetnej forme.
- Aké vlastnosti grafov, alebo akú generatívnu silu zaručujú globálne resp. lokálne použité postupnosti pravidiel?
- Ak prepis pomocou regulárnej gramatiky riadenia nie je schopný vynútiť postupnosť pravidiel na konkrétnom mieste vo vetnej forme, vieme túto postupnosť nahradiť inak?
- Aké triedy grafov vedia, resp. nevedia blokujúce hrany generovať (hor-
né ohraničenie prepisovacieho systému)? Jazyk všetkých grafov nad danou abecedou uzlov a hrán viem generovať aj bez blokujúcich hrán. Existuje jazyk, pre ktorý platí veta 4.1.1 a zároveň nepatrí do triedy blokujúcich edNCE (ak áno, ako vyzerá)?

Hierarchie edNCE jazykov

Okrem klasických hierarchií, ktoré tvoria vnorenie, alebo špeciálne prípady bezkontextových grafových gramatík C-edNCE, existuje hierarchia aj pre nadtriedy C-edNCE: napríklad triedy jazykov edNCE s blokujúcimi hranami a edNCE bez blokujúcich hrán.

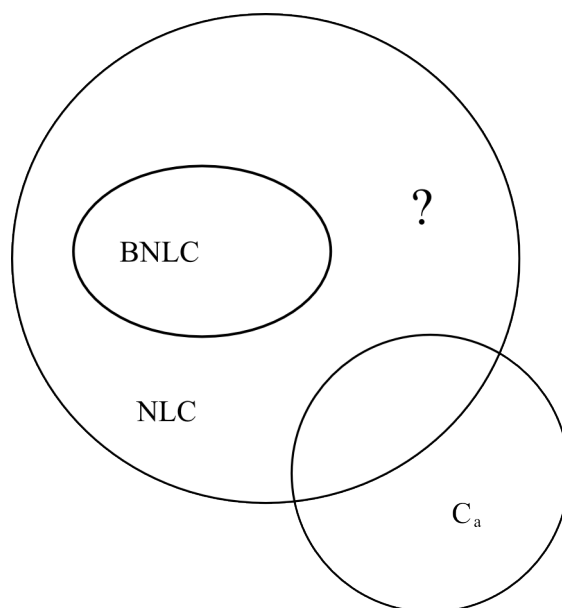
Svojimi vlastnosťami sa neblokujúce a blokujúce edNCE gramatiky a nimi generované jazyky blížia bezkontextovému charakteru triedy C-edNCE a teda vlastnostiam podobným bezkontextovým jazykom nad reťazcami. Problémom sa však ukazuje byť, ako to už bolo naznačené vo vete 4.1.1 o medzere, rozloženie hrán resp. variabilita rozloženia hrán. Variabilita rozloženia hrán naznačuje existenciu nebezkontextových vlastností grafového jazyka vyplývajúcich z existencie nekonečnej MSO definovateľnej podpostupnosti, ktorej rozdiely (medzery) medzi susednými slovami sú väčšie ako konštanta, čiže nie je bezkontextová. Táto vlastnosť neblokujúcich edNCE resp. blokujúcich edNCE gramatík a jazykov sa dá chápať ako akási nepravidelnosť postupností slov a jazykov generovaných týmito gramatikami. V C-edNCE jazykoch a gramatikách ju potláča vlastnosť konfluently. Vlastnosť konfluently je pre potlačenie nebezkontextových vlastností a aj tejto nepravidelnosti, zatiaľ najslabším obmedzením pre edNCE prepisovací systém. C-edNCE nepriamo zakazuje výskyt množstva slov s rovnakou dĺžkou. C-edNCE prepisovací systém zatiaľ definuje najväčšiu triedu bezkontextových grafových jazykov z edNCE.

6.1 Vlastnosti hierarchie edNCE

Rozdiel medzi jazykmi generovanými blokujúcimi edNCE resp. C-edNCE gramatikami sa dá popísať a vidieť celkom zreteľne pomocou nepravidielnosti v raste počtu uzlov špeciálnych jazykov, ako bol napríklad jazyk pre K_n s $n = 2^m$ popísaný vo vete 4.1.5. Nepravidielnosť v tomto jazyku bola spôsobená exponenciálnym rastom počtu uzlov susedných slov tohto jazyka zoradených do usporiadanej postupnosti. Rovnakým spôsobom ako vo vete 4.1.5 vieme odlíšiť aj jazyky generované blokujúcimi edNCE a neblokujúcimi edNCE gramatikami.

Pre neblokujúce edNCE a C-edNCE gramatiky je situácia celkom iná. Obe triedy gramatík sa vyznačujú pravidlenosťou rastu počtu uzlov v nimi generovaných susedných slov jazyka usporiadaných podľa počtu uzlov. Pre obe triedy jazykov platí veta 4.1.1 o medzere medzi susednými slovami daného jazyka. Vzájomným odlíšením sa tu stáva nepravidielnosť, alebo schopnosť neblokujúcej edNCE gramatiky generovať veľké množstvo grafov rovnakého počtu uzlov (z tohto vyplýva akási variabilita hrán v slovách takéhoto jazyka). Ako príklad môže slúžiť jazyk všetkých grafov nad danými abecedami uzlov a hrán $[GR_{\Delta, \Omega}]$, ktorý nie je možné generovať C-edNCE gramatikou, a naopak neblokujúcou edNCE gramatikou áno. Problémom tejto podtriedy jazykov tvoriacej komplement C-edNCE jazykov vzhľadom na neblokujúce edNCE jazyky je, že neexistuje jej presná charakterizácia a popis. Preto odlíšenie C-edNCE a neblokujúcich edNCE jazykov nie je ťažko nahliadnuteľné, z čoho vyplývajú aj následné problémy pri dôkazoch tvrdení o týchto hierarchiách jazykov.

Pre C-edNCE aj neblokujúce edNCE jazyky platí veta 4.1.1, preto medzi každou dvojicou grafov, susedných podľa počtu uzlov, sú konštantné rozdiely v závislosti na gramatike. Podobne aj počty hrán v najbližších slovách neblokujúceho edNCE jazyka nemôžu rásť exponenciálne, lebo počet hrán v grafoch bez multihrán závisí na raste počtu uzlov. Rastú maximálne lineárne.



Obrázok 6.1: Otázka vlastností NLC jazykov, ktoré nie sú BNLC

6.1.1 Otvorené otázky:

- Ako vyzerajú jazyky, ktoré nie sú bezkontextové (C-edNCE), ale dokážeme ich generovať neblokujúcimi edNCE gramatikami? Intuitívne, a na základe predošlých poznatkov sa tieto jazyky môžu líšiť len rozložením hrán v slovách a v počte slov s rovnakou veľkosťou.
- Dokážu neblokujúce edNCE gramatiky generovať grafy s nadmerne nepravidelným rozložením hrán, v zmysle akékoľvek rozloženie hrán a akékoľvek počty slov s rovnakým počtom uzlov? Existujú grafové jazyky, ktoré spĺňajú vetu 4.1.1 a nepatria do neblokujúcich edNCE?

6.2 Vlastnosti hierarchie NLC

Veta 6.2.1. Nech C_a je jazyk množiny všetkých cyklov značených a . Neexistuje BNLC jazyk obsahujúci jazyk C_a

Nie je jasné či existuje NLC jazyk L taký, že nie je v BNLC a neobsahuje C_a . [13]

6.2.1 Zhrnutie:

Všetky doterajšie úvahy a poznatky o hierarchiách grafových gramatík a jazykov vedú k prirodzenej a neprekvapujúcej myšlienke, že zrejme nie je možné zvýšiť generatívnu silu modelu grafových gramatík bez toho aby sme stratili, resp. znížili vhodnosť, zrealizovateľnosť a zložitosť prepisovacieho systému.

Záver

Naším hlavným cieľom bolo skúmanie mier zložitosti, vlastností a porovnanie hierarchií NCE, NLC grafových jazykov a ich rozšírení resp. obmedzení. Práca skúma uvedené ciele z hľadiska generatívneho prístupu skúmania grafových jazykov, ktorý je charakteristický pre prepisovacie systémy gramatík. Definovali sme si základné pojmy a definície z oblasti prepisovania grafov a grafových jazykov, ako boli napríklad blokujúce hrany a konfluencia. Spomenuli sme rôzne spôsoby prepisovania grafov gramatikami a ich odlišnosť v porovnaní s prepismi jazykov nad reťazcami. Bližšie sme si charakterizovali NCE a NLC prepisovacie systémy grafových gramatík, ich rozšírenia aj obmedzenia. Ukázali sme si existujúce hierarchie NCE a NLC. Následne sme charakterizovali problémy pri ich skúmaní a charakterizácii. Konkrétne, problém odlíšenia jednotlivých tried jazykov, resp. problém charakterizácie štruktúry grafových jazykov. Popísali sme bezkontextové aj nebezkontextové vlastnosti NCE prepisovacieho systému s blokujúcimi hranami, bez blokujúcich hrán a s obmedzením konfluencie, a následne sme ich porovnanali s vlastnosťami bezkontextových jazykov nad reťazcami. Pre C-edNCE gramatiky sme si uviedli rôzne spôsoby ich charakterizácie. Z MSO charakterizácie sme predstavili niektoré predošlé výsledky a ukázali sme niektoré ich dôsledky (napr. charakterizáciu štruktúry C-edNCE pomocou nekonečných MSO definovateľných podpostupností a uzáverové vlastnosti). Pri skúmaní vlastností jednotlivých jazykov a grafových gramatík sa ukazuje, že zrejme nie je možné

zvýšiť generatívnu silu modelu grafových gramatík bez toho aby sme stratili, resp. znížili vhodnosť, zrealizovateľnosť a zložitosť prepisovacieho systému. Pre C-edNCE, edNCE bez blokujúcich a edNCE s blokujúcimi hranami sme porovnali ich štruktúru z pohľadu počtu uzlov, hrán a max. stupňa uzlov pre susedné slová v usporiadanej postupnosti jazyka generovaného daným typom gramatiky. Snažili sme sa aj o charakterizáciu štruktúry C-edNCE a generatívnej sily grafových gramatík pomocou vlastností koordinácie prepisu rôznych uzlov vo vetnej forme, hustoty grafov s rovnakým počtom uzlov a hustoty grafov v samotnom jazyku C-edNCE grafových gramatík. Tie isté vlastnosti sme skúmali aj v edNCE bez resp. s blokujúcimi hranami. Schopnosť koordinácie prepisu vetnej formy je charakteristická pre všetky grafové gramatiky so značením resp. preznačovaním hrán. Dospeli sme k nasledujúcim výsledkom: Trieda C-edNCE jazykov pozostáva z jazykov, ktoré majú pomerne veľkú hustotu slov s rôznym počtom uzlov. Rozdiely v počte uzlov medzi veľkosťou (počet uzlov) susednými slovami sú ohraničené konštantou závislou na gramatike. Jazyky C-edNCE sa vyznačujú malou hustotou slov s rovnakou dĺžkou. Slovo rovnakej dĺžky je asymptoticky maximálne $\binom{n+k-1}{k-1}$, kde k je konštanta závislá na parametroch gramatiky. Trieda edNCE jazykov bez blokujúcich hrán je rovnako ako C-edNCE vyznačuje veľkou hustotou slov s rôznym počtom uzlov, ale na rozdiel od C-edNCE, edNCE jazyk môže mať veľkú hustotu slov s rovnakým počtom uzlov (napr. jazyk všetkých grafov nad danými abecedami uzlov a hrán). Jazyk z triedy edNCE jazykov s blokujúcimi hranami môže mať malú hustotu slov s rôznym počtom uzlov (exponenciálne rozdiely medzi susednými slovami) a veľkú hustotu slov rovnakej dĺžky. Ďalej sme sa pokúsili o porovnanie blokujúcich hrán a riadenia odvodenia pomocou regulárnej gramatiky nad C-edNCE a edNCE bez blokujúcich hrán. Nad C-edNCE obe rozšírenia prepisovacieho systému nezvýšia jeho generatívnu silu a nad neblokujúcim edNCE sme dospeli k výsledku, že model s blokujúcimi hranami je buď silnejší ako model s regulárnym riadením odvodenia, alebo sú oba neporovnateľné.

Problémy pri skúmaní vlastností grafových gramatík

Vo všeobecnosti najväčší problém pri skúmaní grafových gramatík tvoril po-

pis ich štruktúry. Pri skúmaní hierarchií NLC a NCE prepisovacích systémov sme narazili na problém charakterizácie ich jazykov, ktoré tvoria odlíšenie jednotlivých hierarchií. Napríklad popis štruktúry jazykov nepatriacich do C-edNCE a patriacich do edNCE bez blokujúcich hrán, ktoré tvoria vzájomné odlíšenie spomenutých tried. Pri skúmaní vlastností gramatík s blokujúcimi hranami a nimi generovaných jazykov sa ukázal ako najväčší problém, že blokujúca hrana neovplyvňuje odvodenie gramatiky priamo svojou existenciou, ale možnosťou svojho vzniku. Blokujúca hrana totiž vynucuje postupnosti pravidiel na určitých miestach vetnej formy možnosťou svojho vzniku pri nesprávnom prepise (použití pravidla) daného miesta. Ďalším problémom je použitie množiny spájacích inštrukcií pravidiel v odvodení vetnej formy. Pri prepise vetnej formy nemusí byť vždy využitá celá množina inštrukcií pravidiel. Z toho dôvodu môže byť výsledok použitia pravidla závislý na tvare prepisovanej časti vetnej formy. Preto vzniká problém pri popise odvodenia v edNCE prepisovacieho systému.

Pokračovanie:

Možným rozšírením tejto práce je dokončenie porovnania blokujúcich hrán a riadenia odvodenia pomocou regulárnej gramatiky. Ďalej charakterizovanie vlastností prepisovacieho systému, spĺňajúceho vlastnosti z vety 4.1.1 o medzere.

Otázky:

- Zvýši riadenie odvodenia regulárnou gramatikou generatívnu silu edNCE?
- Rozširujú blokujúce hrany silu prepisovacieho systému len o jazyky grafov, ktorých rozdiely medzi susednými slovami narastajú lineárne, alebo exponenciálne?
- Blokujúce hrany vynucujú postupnosti pravidiel na určitých miestach vetnej formy. V opačnom prípade by nevzniklo slovo z jazyka, lebo blokujúca hrana sa z vetnej formy nedá odstrániť ani prepísať. Pri porušení týchto postupností pravidiel resp. ich porušením pri odvodení

nejakej konkrétnej časti vetnej formy vzniká blokujúca hrana. Týchto postupností je konečný počet (až na ich nekonečné opakovania vo vetnej forme) a každá z nich má konečnú dĺžku. Ich počet je závislý len na parametroch gramatiky (určite ich nie je viac ako $2^{|P|} * K$). Dajú sa nejakým spôsobom simulovať tieto postupnosti pomocou regulárneho riadenia?

- Regulárne gramatiky riadenia odvodenia generujú regulárne postupnosti pravidiel, ktoré nútia gramatiku generovať odvodenie, v ktorom sa použije rovnaká postupnosť pravidiel. Rozdiel medzi riadením odvodenia regulárnou gramatikou a blokujúcimi hranami je v tom, že riadiaca gramatika vynucuje globálnu postupnosť pravidiel vedúcu k odvodeniu slova a blokujúca hrana vytvára postupnosti pravidiel pre konkrétne miesta vetnej formy definované určitými uzlami a hranami.
- Každá konečná postupnosť je generovateľná regulárnym jazykom, ale vo vetnej forme generovanej gramatikou s blokujúcimi hranami môžu tieto postupnosti vznikáť na rôznych miestach, pričom počet týchto miest môže rásť spolu s vetnou formou (počet miest môže rásť spolu s vetnou formou).
- Simulácia grafových gramatík s blokujúcimi hranami pomocou grafových gramatík s regulárnym alebo iným riadením odvodenia, a naopak.
- Aké množiny postupností pravidiel dokážu resp. nedokážu vynútiť blokujúce hrany?
- Pri akých obmedzeniach edNCE gramatík nebudú blokujúce hrany zvyšovať ich generatívnu silu?

Literatúra

- [1] F. Brandenburg and K. Skodinis. Finite graph automata for linear and boundary graph languages, 2005.
- [2] Diane J. Cook and Lawrence B. Holder. Substructure discovery using minimum description length and background knowledge. *Journal of Artificial Intelligence Research*, 1:231–255, 1994.
- [3] Bruno Courcell. *An axiomatic definition of context-free rewriting and its application to NLC graph grammars*. Springer Berlin / Heidelberg, 1988.
- [4] Bruno Courcelle and Stephan Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101(1-3):77–114, 2000.
- [5] J. Gonzalez, I. Jonyer, L. Holder, and D. Cook. Efficient mining of graph-based data, 2000.
- [6] J.A. Makowsky and J.P. Marino. Treewidth and the monadic quantifier hierarchy.
- [7] Grzegorz Rozenberg. *Handbook of graph grammars and computing by graph transformation*. World Scientific Publishing, 1997.
- [8] K. Skodinis and E. Wanke. Degree problems for finite graph grammars.

- [9] Konstantin Skodinis. The bounded tree-width problem of context-free graph languages. In *Workshop on Graph-Theoretic Concepts in Computer Science*, pages 303–317, 1997.
- [10] Konstantin Skodinis and Egon Wanke. Emptiness problems of eNCE graph languages. *Journal of Computer and System Sciences*, 51(3):472–485, 1995.
- [11] A. O. Slisenko. Computational complexity of string and graph identification. In *MFCS*, pages 182–190, 1979.
- [12] A. O. Slisenko. Context-free grammars as a tool for describing polynomial-time subclasses of hard problems. *Inf. Process. Lett.*, 14(2):52–56, 1982.
- [13] Koichi Yamazaki. *The generating power of the boundary NLC graph grammars*. ACM Annual Computer Science Conference, 1993.