Curs 12

Algoritmi de decizie pentru limbaje independente de context (CFG)

Teoremă:

Este decidabil dacă limbajul lui L(G) pentru G dat este:

(a) vid

(b) finit

(c) infinit

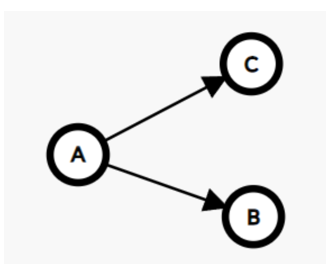
Demonstrație:

Fie $G = (N, T, S, P)$ o gramatică independentă de context.

Pentru a), în forma normală Chomsky se identifică neterminalele folositoare (accesibile din S și terminating). Dacă S este terminating, atunci $L(G) \neq \emptyset$.

Pentru b) și c) transformăm G în gramatica G' în forma normală Chomsky cu $L(G') = L(G) - \{\lambda\}$. $G' = (N', T, S, P')$.

Considerăm un graf orientat pentru G'. Nodurile sunt neterminalele din $G'(N')$. Pentru $A, B \in N'$, avem muchia $(A, B) \iff \exists A \to BC$ sau $A \to CB$ în P'.

Exemplu: 

Demonstrăm că $L(G')$ e finit (b) $\iff$ graful nu are cicluri.

Observație:

G' este în FNC $\Rightarrow$ nu avem simboluri nefolositoare.

Dacă graful are cicluri $\Rightarrow L(G')$ nu este finit.

Luăm un ciclu $A_0A_1...A_nA_0$, deci în gramatică avem: $A_0 \to \alpha_1 A_1 \beta_1 \to \alpha_2 A_2 \beta_2 \to ... \to \alpha_n A_n \beta_n \to \alpha_{n+1} A_0 \beta_{n+1}$ unde $\alpha_i, \beta_i \in N'^*$ cu proprietatea că $|\alpha_i \beta_i| = i$ (din forma normala Chomsky).

G' nu are simboluri nefolositoare $\Rightarrow \alpha_{n+1} \overset{*}{\Rightarrow} w_1$, $\beta_{n+1} \overset{*}{\Rightarrow} w_2$, $w_1, w_2 \in T^*$ și $\exists y, z \in T^*$ astfel încât $S \overset{*}{\Rightarrow} yA_0z$ și $\exists v \in T^*$ astfel încât $A_0 \overset{*}{\Rightarrow} v$. Deci, $S \overset{*}{\Rightarrow} yA_0z, A_0 \overset{*}{\Rightarrow} w_1A_0w_2, A_0 \overset{*}{\Rightarrow} v$. Așadar, $\forall i \geq 1$ $S \overset{*}{\Rightarrow} yA_0z \overset{*}{\Rightarrow} yw_1A_0w_2z \overset{*}{\Rightarrow} yw_1^2A_0w_2^2z \overset{*}{\Rightarrow} yw_1^iA_0w_2^iz \overset{*}{\Rightarrow} yw_1^ivw_2^iz \in L(G') = L(G) - \{\lambda\}$

Demonstrăm că dacă graful nu are cicluri $\Rightarrow L(G')$ e finit. Presupunem că graful nu are cicluri, definim $rank(A) =$ cel mai lung drum în graf care pleacă din $A (A \in N')$. Este bine definit pentru că nu avem drumuri infinite (fără cicluri).

Observație:

Dacă $A \to BC \Rightarrow rank(A) > rank(B)$ și $rank(A) > rank(C)$. Demonstrăm prin inducție că din A nu putem deriva șiruri mai lungi de 2^r unde $r = rank(A)$.

$r = 0 \Rightarrow rank(A) = 0 \Rightarrow outdegree(A) = 0$. CNF ne spune că A este folositor, deci avem producții $A \to a$, a $\in T \Rightarrow$ orice șir derivat are lungimea $1 = 2^0$.

Presupunem că $r > 0$ și proprietatea adevărată pentru neterminalele de $rank < r$. Fie $A \in N'$ cu $rank(A) = r$. Fie $w \in T^*$ astfel încât $A \overset{*}{\Rightarrow} w$. A se rescrie în w cu anumite producții.

Dacă prima derivare este $A \to a, a \in T \Rightarrow |w| = 1$. Dacă primul pas este $A \to BC$ cu $rank(B) < r$ și $rank(c) < r \Rightarrow w = w_1w_2, B \overset{*}{\Rightarrow} w_1, C \overset{*}{\Rightarrow} w_2$ și $|w_1| \leq 2^{r-1}, |w_2| \leq 2^{r-1} \Rightarrow |w| = |w_1| + |w_2| \leq 2^{r-1} + 2^{r-1} = 2^r$.

În gramatica G luăm S (neterminalul de rank maxim). $rank(S) = r_0 \Rightarrow \forall w \in T^*$ derivat din S ($S \overset{*}{\Rightarrow} W$) avem că $|w| \leq 2^{r_0} \Rightarrow L(G')$ e finit. $\mathcal{QED}$

Teoremă:

Pentru cuvântul w și gramatica $G = (N, T, S, P)$ este decidabil dacă $w \in L(G)$.
Demonstrație: algoritmul Cocke–Younger–Kasami cu complexitatea $\mathcal{O}(n^3)$.

Problema corespondenței lui Post (PCP)

Se dau două liste de cuvinte $A = x_1, x_2, ..., x_k$ și $B = y_1, y_2, ..., y_k$ cu $x_i, y_i \in \Sigma^*$,$\forall i = \overline{1, k}$. Spunem că avem o soluție dacă există o secvență de numere $i_1, i_2, ..., i_n$ cu $n \geq 1$ și $i_j \in \{1, 2, .., k\} \forall j = \overline{1, n}$ astfel încât $x_{i_1}x_{i_2}...x_{i_n} = y_{i_1}y_{i_2}...y_{i_n}$

Exemplu: $A = \{1, 10111, 10\}, B = \{111, 10, 0\}$. O soluție este $\{2, 1, 1, 3\}$. A : 10111 1 1 10 = 101111110 , B = 10 111 111 0 = 101111110.

Exemplu: $A = \{10, 011, 101\}, B = \{101, 11, 011\}$ NU are soluție.

Teoremă:

PCP nu este decidabilă.

Reformulare: nu există un algoritm care să decidă dacă PCP are sau nu soluție (chiar și pentru cazul liniar $\Sigma = \{a, b\}$).

Folosim PCP pentru a arăta că anumite proprietăți ale CFG nu sunt decidabile.

Teoremă:

Nu se poate decide algoritmic dacă o gramatică independentă de context este ambiguă.

Demonstrație: Fie $A = \{x_1, x_2, ..., X_k\}$ și $B = \{y_1, y_2, ..., y_k\}$ o instanță a PCP. Fie $\$, 0, 1 \notin \Sigma$ trei simboluri noi. Fie $L_A = \{x_{i_1} x_{i_2} ... x_{i_n} \$01^{i_1}01^{i_2}...01^{i_n} | 1 \leq i_1, i_2, ..., i_n \leq k\}$,și $L_B = \{y_{i_1} y_{i_2} ... y_{i_n} \$01^{i_1}01^{i_2}...01^{i_n} | 1 \leq i_1, i_2, ..., i_n \leq k\}$. Fie gramatica $G = (\{S, S_a, S_b\}, \Sigma \cup \{\$, 0, 1\}, S, P)$ unde $P = \{$

$S \rightarrow S_A | S_B$,

$S_A \rightarrow x_i S_A 01^i | \$ \ \forall i \in \{1..k\}$,

$S_B \rightarrow y_i S_B 01^i | \$ \ \forall i \in \{1..k\}\}$.

Evident, G este independentă de context și $L(G) = L_A \cup L_B$ instanța $PCP(A, B)$ are soluție $\iff \exists i_1, i_2, ... i_n$ astfel încât $x_{i_1} x_{i_2} ... x_{i_n} = y_{i_1} y_{i_2} ... y_{i_n}$. Deci în G producem $x_{i_1} x_{i_2} ... x_{i_n} \$01^{i_n}01^{i_{n-1}}...01^{i_1}$ pe partea S_A și $y_{i_1} y_{i_2} ... y_{i_n} \$01^{i_n}01^{i_{n-1}}...01^{i_1}$ pe partea S_B. Deci avem un cuvânt cu două derivări stângi diferite (reguli liniare). $S \Rightarrow S_A \Rightarrow x_{i_1} S_A 01^{i_1} \Rightarrow x_{i_1} x_{i_2} S_A 01^{i_2} 01^{i_1} \overset{*}{\Rightarrow} x_{i_1} x_{i_2} ... x_{i_n} \$01^{i_n}...01^{i_1}$ analog pe partea S_B, deci G este ambiguă dacă $PCP(A, B)$ are soluție .

Reciproc, arătăm că dacă G este ambiguă, $PCP(A, B)$ are soluție. Deoarece cuvintele 01^i de la dreapta lui \$ ne spun exact ce producții s-au folosit este evident că din S_A (sau S_B) avem o derivare unică. G este ambiguă, deci ambiguitatea provine din alegerea S_A sau S_B la primul pas. Deci există cuvântul W_{AB} astfel încât $S_A \overset{*}{\Rightarrow} W_{AB}$ și $S_B \overset{*}{\Rightarrow} W_{AB}$.

Descompunem W_{AB} și aflăm $i_1, i_2, ..., i_n$ a.î. $x_{i_1} x_{i_2} ... x_{i_n} = y_{i_1} y_{i_2} ... y_{i_n}$. Deci am găsit o soluție pentru $PCP(A, B)$. $\mathcal{QED}$

Probleme nedecidabile:

G_1 și G_2 gramatici independente de context, R expresie regulată. Următoarele întrebări sunt nedecidabile:

(a) G_1 este ambiguă?

(b) $L(G_1) \cap L(G_2) \neq \emptyset$?

(c) $L(G_1) \neq L(G_2)$?

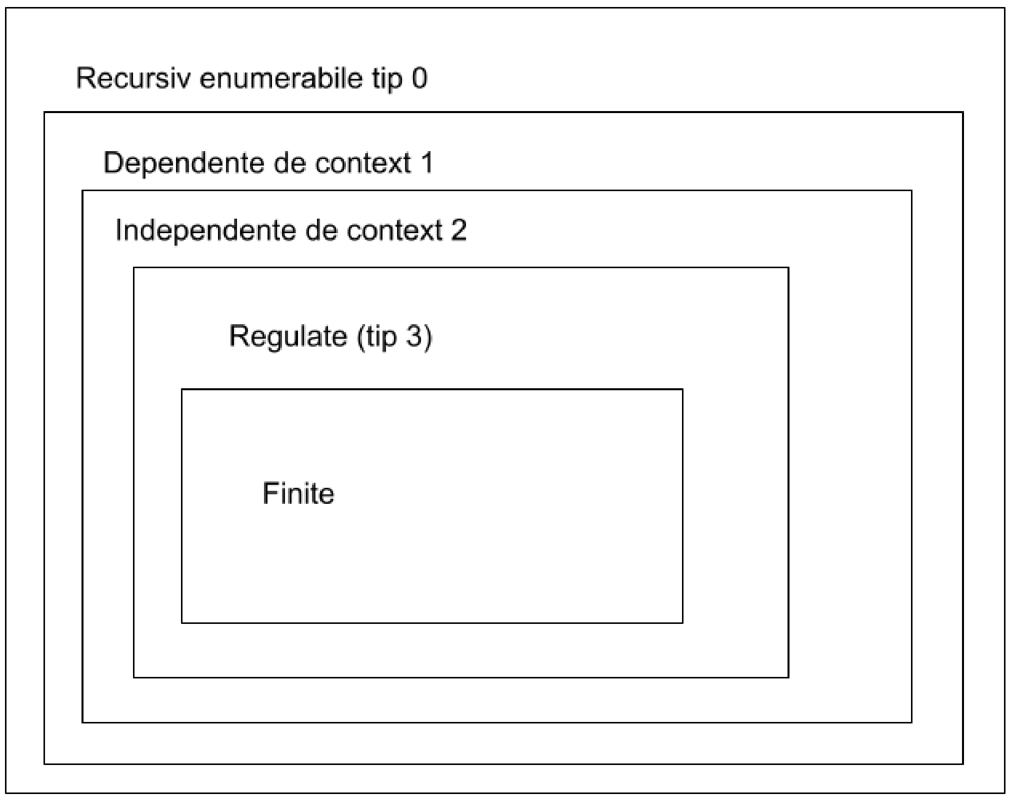
(d) $L(G_1) \neq L(R)$?

(e) $L(G_1) \neq T^*$?

(f) $L(G_2) - L(G_1) \neq \emptyset$?

(g) $L(R) - L(G_1) \neq \emptyset$?

Ierarhia lui Chomsky



Limbajul	Gramatica	Automatul/mașina
Reg. tip 3	regulată	DFA/NFA/$\lambda - NFA$/RE
CF tip 2	CFG	PDA
CS tip 1	CSG	TM linear bounded
RE tip 0	arbitrară	TM