§4 Języki formalne i Gramatyki

- Gramatyki opisują (generują) języki formalne $A \subseteq \Sigma^*$
- Gramatyki różnych stopni zaawansowania
- Problemy rozstrzygalne i nie rozstrzygalne dla różnych typów gramatyk

4.1 Definicje

Gramatyka G zawiera reguły P oraz symbol startowy (axiom) S. Język $L(G)$ generowany przez G, to wszystkie słowa $w \in \Sigma^*$, które można wywodzić z S używając reguły z P.

Uwaga: $S \notin \Sigma$

Def.

(i) Gramatyka G, to 4-elementowa krotka

$$G = (N, \Sigma, P, S)$$

gdzie N i Σ alfabety, takie że $N \cap \Sigma = \emptyset$;

$P \subseteq V^* N V^* \times V^*$ jest skończonym zbiorem reguł ($V = N \cup \Sigma$). piszemy $u \to v$ dla $(u,v) \in P$

$S \in N$ jest symbolem startowym

(ii) $x, y \in V^*$

$x \Rightarrow_G y$, jeżeli y można wywodzić z x w jednym kroku, czyli

$x \Rightarrow_G y$, jeżeli istnieją $w, w' \in V^*$ oraz $u \to v \in P$, takie że $x = wuw'$ i $y = wvw'$.

(iii) $A \in N$

$L(A, G) := \{ w \in \Sigma^* \mid A \overset{*}{\Rightarrow}_G w \}$

$L(G) := L(S, G)$

(iv) Gramatyki G_1 i G_2 są równoważne, jeżeli $L(G_1) = L(G_2)$.

Przykłady:

(i) $G_1 = (N_1, \Sigma_1, P_1, S)$, $\Sigma_1 = \{p, q, +, -, (,)\}$, $N_1 = \{S\}$

$P_1 = \{ S \to S+S, S \to S-S, S \to (S), S \to p, S \to q \}$

$S \Rightarrow S-S \Rightarrow p-S \Rightarrow p-(S) \Rightarrow p-(S+S)$

$\Rightarrow p-(q+S) \Rightarrow p-(q+p)$

czyli $p-(q+p) \in L(G_1)$

(ii) $G_2 = (N_2, \Sigma_2, P_2, S)$, $\Sigma_2 = \{a, b\}$, $N_2 = \{S, T\}$

$P_2 = \{ S \to aS \mid aT, T \to bT \mid b \}$

$S \Rightarrow aS \Rightarrow aaS \Rightarrow aaaT \Rightarrow aaabT \Rightarrow aaabb$

$L(G_2) = \{ a^i b^j \mid i, j \geq 1 \}$

(iii) $G_3 = (N_3, \Sigma_3, P_3, S)$, $\Sigma_3 = \{a, b\}$, $N_3 = \{S\}$

$P_3 = \{ S \to aSb \mid \lambda \}$

$L(G_3) = \{ a^n b^n \mid n \geq 0 \}$

(iv) Gramatyka G_4, taka że $L(G_4) = \{a^n b^n c^n \mid n \geq 1\}$ potrzebuje "dodatkowy" $C \in N$, który "pamięta", ile c muszą być w słowie, czyli generujemy słowa $a^n b (Cb)^{n-1} c$, shiftujemy C na prawo i zmienimy C na c.

$\Sigma_4 = \{a, b, c\}$, $N_4 = \{C, S, T\}$

$P_4 = \{ S \to Tc,\ T \to aTCb \mid ab,$
$Cb \to bC,\ Cc \to cc \}$

$S \Rightarrow Tc \Rightarrow aTCbc \Rightarrow aaTCbCbc$
$\Rightarrow aaaTCbCbCbc \Rightarrow aaaabCbCbCbc$
$\overset{*}{\Rightarrow} aaaabbbbCCCc \overset{*}{\Rightarrow} aaaabbbbcccc$

<u>Def.</u> Hierarchia Chomsky'ego

Gramatyka $G = (N, \Sigma, P, S)$ jest

(i) ogólna (z typu 0), jeżeli nie ma ograniczeń reguł.

(ii) kontekstowa (z typu 1), jeżeli wszystkie reguły w P mają formę

$$xAy \to xzy, \quad A \in N, z \in V^+, x, y \in V^*$$

(iii) bezkontekstowa (z typu 2), jeżeli wszystkie reguły w P mają formę

$$A \to z, \quad A \in N, z \in V^*$$

(iv) prawostronnie liniowa (z typu 3), jeżeli wszystkie reguły w P mają formę

$$A \to aB \text{ lub } A \to a \text{ lub } A \to \lambda,$$

$$A, B \in N, \ a \in \Sigma$$

<u>Def:</u>

Język formalny $A \subseteq \Sigma^*$ jest z typu i $(i = 0, \dots 3)$, jeżeli istnieje gramatyka G z typu i, taka że $A = L(G)$.

<u>Przykład</u>

G_2 jest z typu 0, 1, 2, 3.

G_3 jest z typu 0, 2, ale nie z typu 1, 3.

G_4 jest z typu 0, 1, ale nie z typu 2, 3.

<u>Def:</u>

$$RL := \{ A \subseteq \Sigma^* \mid A \text{ z typu } 3 \}$$

$$CF := \{ A \subseteq \Sigma^* \mid A \text{ z typu } 2 \}$$

$$CS := \{ A \subseteq \Sigma^* \mid A \text{ z typu } 1 \}$$

$$RE := \{ A \subseteq \Sigma^* \mid A \text{ z typu } 0 \}$$

<u>Przykład:</u>

(i) $L = \{ a^i b^j \mid i, j \geq 1 \} \in RL$,

ponieważ G_2 jest z typu 3 i $L(G_2) = L$.

Też $L \in CF$, ponieważ G_2 też z typu 2.

(ii) $L = \{a^n b^n \mid n \geq 0\} \in CF$,

ponieważ G_3 z typu 2 i $L(G_3) = L$.

uwaga: G_3 nie jest z typu 1, ale mimo to $L \in CS$

Twierdzenie

$$RL \subsetneq CF \subsetneq CS \subsetneq RE$$

"Dowód":

$CF \subseteq CS$: $L \in CF$, wtedy $L - \{\lambda\} \in CS$.

$RL \neq CF$, $CF \neq CS$: lematy o pompowaniu.

Twierdzenie

$L_1, L_2 \subseteq \Sigma^*$.

Jeżeli $L_1, L_2 \in RL(CF, CS, RE)$, wtedy

$L_1 \cup L_2$, $L_1 L_2$, L^+, $L^* \in RL(CF, CS, RE)$.

"Dowód":

$G_1 = (N_1, \Sigma, P_1, S_1)$, $G_2 = (N_2, \Sigma, P_2, S_2)$, takie że

$L_1 = L(G_1)$, $L_2 = L(G_2)$ i $N_1 \cap N_2 = \emptyset$

$G_0 := (N_0, \Sigma, P_0, S_0)$

$S_0 \notin N_1 \cup N_2$, $N_0 := N_1 \cup N_2 \cup \{S_0\}$

$P_0 := P_1 \cup P_2 \cup \{S_0 \to z \mid S_1 \to z \in P_1 \vee S_2 \to z \in P_2\}$

wtedy $L(G_0) = L_1 \cup L_2$.

Twierdzenie $L \subseteq \Sigma^*$

L przeliczalny wtw. istnieje gramatyka G, taka że $L = L(G)$.

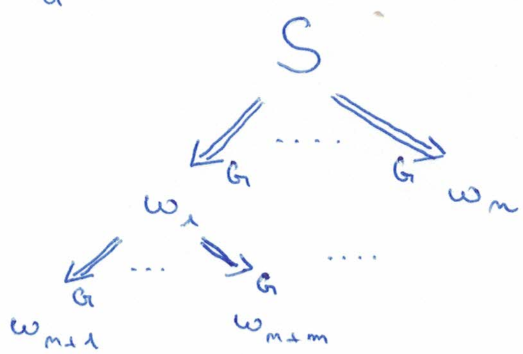
Dowód:

„$\Leftarrow$" Niech $L = L(G)$.

Opisujemy maszynę Turinga M, która obliczy funkcję

$$f(w) = \begin{cases} w; & w \in L(G) \\ \uparrow; & w \notin L(G) \end{cases}$$

M buduje drzewo T wszystkich $w \in V^*$, takich że $S \overset{*}{\Rightarrow}_G w$:



uwaga: Ponieważ zbiór reguł P skończony, jest skończona ilość słów na każdym poziomie T. Jeżeli $w \in L(G)$, wtedy w wystąpi w T i M zwraca w; jeżeli $w \notin L(G)$, wtedy w nie wystąpi w T i M nie zatrzymuje się.

„$\Rightarrow$" Niech M będzie maszyną Turinga, taka że M zatrzyma się z wejściem x wtw. $x \in L$, czyli $x \in L$ wtw. $q_0 b x \overset{*}{\Rightarrow} u q_s v$.

Gramatyka G symuluje M odwrotnie:

$q_0 b x = k_0 \Rightarrow k_1 \Rightarrow \dots k_m = u q_s v$

Pomysł:

$P = P_1 \cup P_2 \cup P_3$, takie że

$S \overset{*}{\Rightarrow}_G [u q_s v]$ używając P_1

$[k_{i+1}] \overset{*}{\Rightarrow} [k_i]$ używając P_2

$[k_0] \overset{*}{\Rightarrow} x$ używając P_3

czyli

$G = (N, \Sigma, P, S)$

$N = \Sigma_0 \cup \{ [,], T_1, T_2, T_3, S \} \cup Q$

$P = P_1 \cup P_2 \cup P_3$

$P_1 = \{ S \to [q_n], q_n \to a_i q_n \mid q_n a_i \}$

P_2: Niech $T(q_i, a) = (\sigma, q_j)$, czyli n.p.

$u q_i a v = k_e \Rightarrow k_{e+1} = u a q_j v$.

wtedy $u a q_j v \to u q_i a v \in P_2$

P_3: $[b^s q_0 b x b^t] \overset{*}{\Rightarrow}_G T_2 x b^t]$

$\overset{*}{\Rightarrow}_G x T_3 b^t] \overset{*}{\Rightarrow}_G x$

czyli

$P_3 = \{ q_0 b \to T_1, a_0 T_1 \to T_1, [T_1 a_0 \to T_2,$

$T_2 a_i \to a_i T_2, T_2 \to T_3,$

$T_3 b \to T_3, T_3] \to \lambda \}$

wtedy

$x \in L(G)$ wtw. $q_0 b x \overset{*}{\Rightarrow} u q_s v$ wtw. $x \in L$.

Def: G gramatyka, $w \in \Sigma^*$

(i) Problem słowa

$P(G, w)$ wtw. $w \in L(G)$.

(ii) Specjalny problem słowa

$P_G(w)$ wtw. $w \in L(G)$.

Twierdzenie

(i) $P_G(w)$ nie jest rozstrzygalny.

(ii) $G \in RL, CF, CS$

$P_G(w)$ jest rozstrzygalny.

Twierdzenie $RE \neq CS$

Twierdzenie G gramatyka

Nie rozstrzygalny są

(i) $L(G) = \emptyset$

(ii) $L(G)$ skończony

(iii) $L(G) = \Sigma^*$

(iv) $x_0 \in L(G)$

Dowód:

$M = \{x \mid Def(\phi_x) = \emptyset\}$ nie jest roztrzygalny.

obliczymy dla każdego $x \in \Sigma^*$ gramatykę G_x,

taką że $x \in M$ wtw. $L(G_x) = \emptyset$

$(\text{czyli } M \leq \{G \mid L(G) = \emptyset\})$

Jeżeli $x \in \Sigma^*$ jest kodowaniem maszyny Turinga, G_x będzie gramatyką z ostatniego dowodu; wtedy $L(G_x) = Def(\phi_x)$.

Jeżeli $x \in \Sigma^*$ nie jest kodowaniem maszyny Turinga $G_x := (N, \Sigma, \emptyset, S)$; wtedy $L(G_x) = \emptyset$.

czyli $f(x) := G_x$ obliczalna oraz

$x \in M$ wtw. $Def(\phi_x) = \emptyset$ wtw. $G_x \in \{G \mid L(G) = \emptyset\}$