4. Limbaje regulate. Lema stelei

Un limbaj $\boldsymbol{L}$ este regulat dacă este generat de o gramatică regulată $\boldsymbol{G}$.

Lema stelei. Fie $\boldsymbol{L}$ un limbaj regulat. Atunci există $n \in \mathbb{N}$, astfel încât oricare ar fi $w \in \boldsymbol{L}$ cu $|w| \geq n$ poate fi scris/descompus sub forma *w = xyz*, unde:

1) $|xy| \leq n$,

2) $0 < |y| \leq n$, $\qquad (*)$

3) $xy^k z \in L, \forall k \in \mathbb{N}$.

Demonstrație. Conform consecinței 3.1. se poate construi automatul finit determinist $\boldsymbol{A_D} = (\boldsymbol{\Sigma}, \boldsymbol{Q}, \boldsymbol{\delta}, \boldsymbol{q_0}, \boldsymbol{F})$ echivalent cu gramatica regulată $\boldsymbol{G}$ care generează limbajul regulat $\boldsymbol{L}$, adică:

$$\boldsymbol{\mathcal{T}}(\boldsymbol{A_D}) = \boldsymbol{L}.$$

Fie *n* = card $\boldsymbol{Q}$ = $|\boldsymbol{Q}|$ şi fie $w \in \boldsymbol{L}$ cu $|w| \geq n$, adică w este acceptat de automatul $\boldsymbol{A_D}$. Atunci de la starea q_0 la starea $\boldsymbol{\delta}(q_0, w) \in \boldsymbol{F}$, automatul trece prin cel puțin *n + 1* stări.

Se notează prin q^1 stările cele mai apropiate care se repetă. Atunci stările prin care trece automatul se pot reprezenta astfel:

$$q_0 \to \cdots \to q^1 \to \cdots \to q^1 \to \cdots \to \boldsymbol{\delta}(q_0, w).$$

Alegând cuvântul *x* astfel încât $\boldsymbol{\delta}(q_0, x) = q^1$, cuvântul *y* astfel încât $\boldsymbol{\delta}(q^1, y) = q^1$ şi cuvântul *z* astfel încât $\boldsymbol{\delta}(q^1, z) = \boldsymbol{\delta}(q_0, w)$ se constată că:

- *w = xyz*;
- $0 < |y| \leq n$;
- pentru *k* = 0 are loc *x z* ∈ $\boldsymbol{L}$, deoarece

$$\boldsymbol{\delta}(q_0, \boldsymbol{x}z) = \boldsymbol{\delta}(\boldsymbol{\delta}(q_0, x), z) = \boldsymbol{\delta}(q^1, \mathrm{z}) =$$
$$= \boldsymbol{\delta}(q_0, w) \in \boldsymbol{F};$$

- pentru *k* ≥ 2 are loc $xy^k z \in \boldsymbol{L}$, deoarece

$$\boldsymbol{\delta}(q_0, \boldsymbol{x}y^k z) = \boldsymbol{\delta}(\boldsymbol{\delta}(q_0, x), y^k z) = \boldsymbol{\delta}(q^1, y^k \mathrm{z}) = \boldsymbol{\delta}(\boldsymbol{\delta}(q^1, y), y^{k-1}\mathrm{z}) = \cdots =$$
$$= \boldsymbol{\delta}(\boldsymbol{\delta}(q^1, y), \mathrm{z}) = \boldsymbol{\delta}(q^1, z) = \boldsymbol{\delta}(q_0, w) \in \boldsymbol{F}.$$

Prin urmare, lema stelei este adevărată.

Lect. univ. dr. TATIANA-CORINA DOSESCU
Facultatea de Informatică
Universitatea Titu Maiorescu

Aplicație. Folosind lema stelei să se arate că limbajul $\boldsymbol{L} = \{a^m b^m | m \geq 1\}$ nu este un limbaj regulat.

Rezolvare. Presupunem prin reducere la absurd că ***L*** este limbaj regulat. Atunci, conform lemei stelei există *n* ∈ *ℕ*, astfel încât oricare ar fi *w* ∈ ***L*** cu $|w| \geq n$ poate fi scris sub forma *w = x y z*, unde:

1) $|xy| \leq n$,

2) $0 < |y| \leq n$, $\qquad (*)$

3) $xy^k z \in \boldsymbol{L}, \forall k \in \mathbb{N}$.

Fie *w* = $a^m b^m \in \boldsymbol{L}$ cu $|w| = 2m \geq n$, atunci $a^m b^m = xyz$ cu $|xy| \leq n$ şi $|y| \neq 0$.

Sunt posibile următoarele cazuri:

i) *xy* este format numai din simboluri *a.* Atunci *y* conține cel puțin un simbol *a*, iar *x* are cel mult *m-1* de simboluri *a*. În acest caz, pentru *k* = 0 se obține *x z* = $a^m b^m$∈ ***L***, adică $|xz| = 2m$. Pe de altă parte, *x* are cel mult *m-1* de simboluri *a*, iar *z* are exact *m* simboluri de *b* și deci $|xz| < 2m$, deoarece $|xz| \leq 2m - 1$. În concluzie, are loc $|xz| = 2m$ şi $|xz| < 2m$, adică 2*m* < 2*m*, contradicție!

ii) *x* are *m* simboluri *a* și $m_1 \neq 0$, $m_1 < m$ simboluri *b*, adică $|x| = m + m_1$, iar *y* conține cel puțin un simbol *b*. În acest caz, pentru *k* = 0 se obține *xz* = $a^m b^m$ ∈ ***L***, adică $|xz| = 2m$. Pe de altă parte, *z* are cel mult $m - m_1 - 1$ simboluri *b* și $|xz| \leq m + m_1 + m - m_1 - 1 = 2m - 1$. În concluzie, are loc $|xz| = 2m$ şi $|xz| < 2m$, adică 2*m* < 2*m*, contradicție!

Pentru a elimina contradicția trebuie renunțat la presupunerea inițială, adică limbajul ***L*** nu este limbaj regulat.

Următoarea propoziție se referă la posibilitatea de a stabili dacă un limbaj regulat este infinit.

Propoziția 4.1. Fie ***G*** o gramatică regulată. Atunci limbajul $\boldsymbol{L(G)}$, generat de gramatică, este infinit, dacă și numai dacă există *w* ∈ $\boldsymbol{L(G)}$ cu *n* ≤ $|w| \leq 2n$, unde *n* este din **lema stelei**.

Demonstrație.

Implicația ($\Leftarrow$) , adică presupunem că există *w* ∈ $\boldsymbol{L(G)}$ cu *n* ≤ $|w| \leq 2n$. Atunci conform lemei stelei există descompunerea:

$$\left[\begin{array}{l} \quad w = xyz \text{ cu:} \\ \quad 0 < |y| \leq n; \\ xy^k z \in \boldsymbol{L(G)}, \forall k \in \mathbb{N}. \end{array}\right.$$

Cum $|y| > 0$, şirul de inegalităţi stricte

$$|xz| < |xyz| < |xy^2 z| < |xy^3 z| \ldots$$

tinde la ∞, adică limbajul $\boldsymbol{L}(\boldsymbol{G})$ este infinit.

Implicația (⇒), adică presupunem că $\boldsymbol{L}(\boldsymbol{G})$ este infinit. Atunci există $w \in \boldsymbol{L}(\boldsymbol{G})$ cu

$n \leq |w|$.

Dacă $|w| \leq 2n$, atunci $n \leq |w| \leq 2n$.

În caz contrar, $|w| > 2n$ şi fie $w^1 = xz \in \boldsymbol{L}(\boldsymbol{G})$, unde $w = xyz$, conform lemei stelei şi $0 < |y| \leq n$. Cum lungimea lui w^1 scade faţă de lungimea lui w cu cel mult n, rezultă

$|w^1| \geq n$ și se repetă raționamentul pentru w^1. Cum lungimea nolui w^1 este finită, după un număr finit de paşi se ajunge la un w cu $n \leq |w| \leq 2n$.

Practic, este suficient să se genereze toate cuvintele $(\boldsymbol{N} \cup \boldsymbol{T})^*$de lungime < $2n$ și se caută printre ele un cuvânt din $\boldsymbol{T}^*$ de lungime ≥ n.

5. Gramatici independente de context

5.1. Recapitulare (din cursul 2)

Gramatica $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ este **independentă de context** dacă orice producție a ei este de forma $A \to \alpha$, unde $A \in \boldsymbol{N}$ și $\alpha \in (\boldsymbol{N} \cup \boldsymbol{T})^*$.

Fie ***G*** o gramatică independentă de context. O derivare a sa este numită **derivare stângă** și este notată prin $\overset{s}{\Rightarrow}$, dacă la fiecare pas se înlocuiește neterminalul cel mai din stânga folosind o producție a lui ***G***.

Într-o gramatică independentă de context ***G*** orice cuvânt $w \in \boldsymbol{L}(\boldsymbol{G})$ se poate obţine printr-o derivare stângă din simbolul inițial ***S***.

5.2. Simplificarea gramaticilor independente de context(I)

Prin **simplificarea unei gramatici independente de context** $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ se înţelege simplificarea producţiilor lui ***G*** prin eliminarea unor simboluri și producții inutile.

Definiția 1. Se numește $\boldsymbol{\lambda}$ - **producție** o producție de forma $A \to \boldsymbol{\lambda}$, cu $A \in \boldsymbol{N}$, iar $\boldsymbol{\lambda}$ este cuvântul nul (cuvântul nul a fost notat și cu $\boldsymbol{\varepsilon}$).

Observația 1. Dacă $\lambda \in \boldsymbol{L}(\boldsymbol{G})$, atunci există cel puțin o λ - producție în ***P***.

Observația 2. Dacă $\boldsymbol{G}$ are o λ - producție, atunci nu rezultă că $\lambda \in \boldsymbol{L}(\boldsymbol{G})$.

Exemplul 1. Fie $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$, unde $\boldsymbol{N} = \{S\}$, $\boldsymbol{T} = \{a\}$ și mulțimea producțiilor $\boldsymbol{P}$ este:

$$S \rightarrow aA \quad (1)$$
$$A \rightarrow aA \quad (2)$$
$$A \rightarrow \lambda \quad (3)$$

Rezultă că $\boldsymbol{L}(\boldsymbol{G}) = \{a^n | \; n \geq 1\}$ și deci $\lambda \notin \boldsymbol{L}(\boldsymbol{G})$.

Teorema 1. Pentru orice gramatică independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$, care conține λ - producții, există o gramatică independentă de context $\boldsymbol{G^1} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P^1})$ echivalentă cu $\boldsymbol{G}$ și care nu conține λ - producții.

Propoziția 1. Pentru a elimina λ - producțiile dintr-o gramatică independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ și a obține gramatica independentă de context $\boldsymbol{G^1} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P^1})$ fără λ - producții, se poate utiliza următorul algoritm:

Pasul 1. Definim o mulțime $\boldsymbol{M} = \left\{A \in \boldsymbol{N} \middle| \; A \overset{*}{\Rightarrow} \lambda\right\}$;

Pasul 2. Definim $\boldsymbol{P^1} = \boldsymbol{P} \backslash \{A \rightarrow \lambda | \; A \in \boldsymbol{M}\}$, numită $\boldsymbol{P^1}$ iniţială;

Pasul 3. Pornind de la $\boldsymbol{P^1}$ inițială, pentru fiecare producție ce conține în membrul drept un neterminal $A \in \boldsymbol{M}$ adăugăm în $\boldsymbol{P^1}$ inițială o nouă producție echivalentă cu cea inițială, prin înlocuirea neterminalului *A* cu λ, obținându-se în acest fel forma finală a lui $\boldsymbol{P^1}$.

Exemplul 2. Pentru a elimina λ - producțiile din gramatica independentă de context din exemplul 1, aplicăm algoritmul prezentat în propoziția 1, astfel:

Pasul 1. $\boldsymbol{M} = \{A\}$;

Pasul 2. $\boldsymbol{P^1} = \boldsymbol{P} \setminus \{A \rightarrow \lambda | \; A \in \boldsymbol{M}\} = \{S \rightarrow aA, A \rightarrow aA\}$;

Pasul 3.

$$\boldsymbol{P^1} = \{S \rightarrow aA, A \rightarrow aA\} \cup \{S \rightarrow a, A \rightarrow a\} = \{S \rightarrow aA, A \rightarrow aA, S \rightarrow a, A \rightarrow a\}$$

este forma finală a lui $\boldsymbol{P^1}$.

Exemplul 3. Fie gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$, unde

$\boldsymbol{N} = \{S, X, Y, Z, W\}$, $\boldsymbol{T} = \{a, b\}$ și mulțimea producțiilor $\boldsymbol{P}$:

$$S \rightarrow X | \; XY | \; Z \quad (1)$$
$$X \rightarrow Z | \; \lambda \quad (2)$$
$$Y \rightarrow Wa | \; a \quad (3)$$
$$Z \rightarrow WX | \; aZ | \; Zb \quad (4)$$
$$W \rightarrow XYZ | \; bXa | \; \lambda \quad (5)$$

Eliminarea λ - producțiilor din gramatica $\boldsymbol{G}$ se realizează cu algoritmul descris în propoziția 1, astfel:

Pasul 1. **M** = $\{S, X, Z, W\}$ deoarece $S \underset{(1)}{\overset{*}{\Rightarrow}} X \underset{(2)}{\overset{*}{\Rightarrow}} \lambda$ şi $Z \underset{(4)}{\overset{*}{\Rightarrow}} WX \underset{(2)}{\overset{*}{\Rightarrow}} W \underset{(5)}{\overset{*}{\Rightarrow}} \lambda$;

Pasul 2. Se determină mulțimea $\boldsymbol{P^1}$ inițială:

$S \rightarrow X \mid XY \mid Z$ (1)

$X \rightarrow Z$ (2)

$Y \rightarrow Wa \mid a$ (3)

$Z \rightarrow WX \mid aZ \mid Zb$ (4)

$W \rightarrow XYZ \mid bXa$ (5)

Pasul 3. Determinăm mulțimea $\boldsymbol{P^1}$ finală, determinând mai întâi producțiile echivalente cu λ - producțiile existente în $\boldsymbol{P^1}$ iniţială:

Producție inițială	**Producții echivalente**
$S \rightarrow X$	$S \nrightarrow \lambda$
$S \rightarrow XY$	$S \rightarrow Y$
$S \rightarrow Z$	$S \nrightarrow \lambda$
$X \rightarrow Z$	$X \nrightarrow \lambda$
$Y \rightarrow Wa$	$Y \rightarrow a$
$Z \rightarrow WX$	$Z \rightarrow X$ $Z \rightarrow W$ $Z \nrightarrow \lambda$
$Z \rightarrow aZ$	$Z \rightarrow a$
$Z \rightarrow Zb$	$Z \rightarrow b$
$W \rightarrow XYZ$	$W \rightarrow YZ$ $W \rightarrow XY$ $W \rightarrow Y$
$W \rightarrow bXa$	$W \rightarrow ba$

Producțiile din coloana a doua, având săgeata tăiată, nu sunt permise, motivul fiind evident.

Mulțimea $\boldsymbol{P^1}$ finală se obține din mulțimea $\boldsymbol{P^1}$ inițială reunită cu mulțimea producțiilor echivalente din tabelul de mai sus, fiind formată din următoarele producții:

$$S \to X|\ XY|\ Z|\ Y \quad (1)$$
$$X \to Z \quad (2)$$
$$Y \to Wa|\ a \quad (3)$$
$$Z \to WX|\ aZ|\ Zb|\ W|\ X|\ a|\ b \quad (4)$$
$$W \to XYZ|\ bXa|\ YZ|\ XY|\ Y|\ ba \quad (5)$$

Definiția 2. Se numește **redenumire** o derivare de forma $A \overset{*}{\Rightarrow} B$, unde $A, B \in \boldsymbol{N}$.

Teorema 2. Fie $\boldsymbol{G}$ o gramatică independentă de context. Atunci există o gramatică independentă de context $\boldsymbol{G^0}$ echivalentă cu $\boldsymbol{G}$ în care nu apar redenumiri.

Demonstrație. Fie gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$. Pornind de la gramatica $\boldsymbol{G}$ se construiește gramatica independentă de context $\boldsymbol{G^0} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P_0})$, unde mulţimea producţiilor $\boldsymbol{P_0}$ se obţine astfel:

$$\boldsymbol{P_0} = \left\{A \to \alpha \middle|\ \exists B \in \boldsymbol{N} \text{ cu } A \overset{*}{\Rightarrow} B \text{ și } B \to \alpha \in \boldsymbol{P_2}\right\}, \boldsymbol{P_2} = \boldsymbol{P} \setminus \boldsymbol{P_1},$$

$$\boldsymbol{P_1} = \{A \to B|\ A, B \in \boldsymbol{N} \text{ și } A \to B \in \boldsymbol{P}\}.$$

Se observă că $\boldsymbol{P_0}$ se obține din $\boldsymbol{P}$ prin eliminarea redenumirilor, care nu afectează statutul de gramatică independentă de context al gramaticii $\boldsymbol{G^0}$.

Trebuie demonstrat acum că $\boldsymbol{L}(\boldsymbol{G}) = \boldsymbol{L}(\boldsymbol{G^0})$, adică echivalenţa dintre cele două gramatici, demonstrând următoarele două incluziuni:

$\boldsymbol{L}(\boldsymbol{G}) \supset \boldsymbol{L}(\boldsymbol{G^0})$ rezultă observând că dacă $\alpha \underset{G^0}{\overset{*}{\Rightarrow}} \beta$ rezultă și că $\alpha \underset{G}{\overset{*}{\Rightarrow}} \beta$:

$\boldsymbol{L}(\boldsymbol{G}) \subset \boldsymbol{L}(\boldsymbol{G^0})$ se obţine astfel:

fie $S \underset{G}{\overset{S}{\Rightarrow}} \propto_1 \underset{G}{\overset{S}{\Rightarrow}} \propto_2 \underset{G}{\overset{S}{\Rightarrow}} \dots \underset{G}{\overset{S}{\Rightarrow}} \propto_{k+1} \underset{G}{\overset{S}{\Rightarrow}} w$ o derivare stângă din $\boldsymbol{G}$ și fie k cel mai mare indice pentru care $S \to \propto_1$, $\propto_1 \to \propto_2$, ... , $\propto_{k-1} \to \propto_k \in \boldsymbol{P_1}$. Atunci $S \to \propto_{k+1} \in \boldsymbol{P_0}$ și deci $S \underset{G}{\overset{S}{\Rightarrow}} w$, adică incluziunea este adevărată.

Propoziția 2. Eliminarea redenumirilor dintr-o gramatică independentă de context se poate realiza folosind următorul algoritm:

Pasul 1. Se determină toate redenumirile din gramatica $\boldsymbol{G}$;

Pasul 2. Pentru fiecare redenumire de forma $A \overset{*}{\Rightarrow} B$ considerăm toate producțiile de forma $B \to \alpha$ și pentru fiecare dintre aceste producții adăugăm în $\boldsymbol{P_0}$ câte o nouă producție de forma $A \to \alpha$.

Pasul 3. Adăugăm în $\boldsymbol{P_0}$ toate producţiile din $\boldsymbol{P}$ care nu sunt redenumiri.

Exemplul 4. Fie gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$, unde

N = $\{S, A, B\}$, ***T*** = $\{a, b\}$ și mulțimea producțiilor ***P*** este următoarea:

$$
\begin{array}{ll}
S \to A & (1)\\
A \to aSB & (2)\\
A \to B & (3)\\
B \to bSA & (4)\\
A \to a & (5)\\
B \to b & (6)
\end{array}
$$

Determinăm toate redenumirile din ***P***, precum și producțiile noi necesare, astfel:

Redenumiri	**Producții noi**
$S \to A$	$S \to aSB$ $S \to a$
$A \to B$	$A \to bSA$ $A \to b$
$S \overset{*}{\Rightarrow} B$	$S \to bSA$ $S \to b$

Gramatica independentă de context echivalentă cu ***G*** și fără redenumiri este

$\boldsymbol{G^0} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P_0})$, unde mulţimea producţiilor $\boldsymbol{P_0}$ este următoarea:

$$
\left.\begin{array}{ll}
A \to aSB & (1)\\
B \to bSA & (2)\\
A \to a & (3)\\
B \to b & (4)
\end{array}\right\} \text{producțiile din } \boldsymbol{P} \text{ care nu sunt redenumiri}
$$

$$
\left.\begin{array}{ll}
S \to aSB & (5)\\
S \to a & (6)\\
A \to bSA & (7)\\
A \to b & (8)\\
S \to b & (9)\\
S \to bSA & (10)
\end{array}\right\} \text{producții noi.}
$$

Exemplul 5. Fie gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$, unde

N = $\{S, X, Y, Z\}$, ***T*** = $\{a, b\}$ și mulțimea producțiilor ***P*** este următoarea:

$$
\begin{array}{ll}
S \to X \mid Y \mid bb & (1)\\
X \to Z \mid aXY & (2)\\
Y \to Xa \mid a & (3)
\end{array}
$$

$$Z \to XY|\ S|\ Zb \quad (4)$$

Pentru a se construi gramatica independentă de context $\boldsymbol{G^0} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P_0})$, echivalentă cu ***G*** și fără redenumiri, aplicând pașii 1 și 2 ai algoritmului de mai sus, mai întâi se determină toate redenumirile din ***P***, precum și producțiile noi corespunzătoare, evitând eventualele redenumiri noi, astfel:

Redenumirile din *P*	**Producțiile noi corespunzătoare**
$S \to X$	$S \to aXY$
$S \to Y$	$S \to Xa|\ a$
$X \to Z$	$X \to XY|\ Zb$
$Z \to S$	$Z \to bb$
$S \overset{*}{\Rightarrow} Z$	$S \to XY|\ Zb$
$X \overset{*}{\Rightarrow} S$	$X \to bb$
$X \overset{*}{\Rightarrow} Y$	$X \to Xa|\ a$
$Z \overset{*}{\Rightarrow} X$	$Z \to aXY$
$Z \overset{*}{\Rightarrow} Y$	$Z \to Xa|\ a$

Conform pasului 3 al algoritmului, $\boldsymbol{P_0}$ se obține adăugându-se la producțiile noi producțiile inițiale care nu sunt redenumiri, astfel:

$$S \to bb|\ aXY|\ Xa|\ a|\ XY|\ Zb \qquad (1)$$
$$X \to aXY|\ XY|\ Zb|\ bb|\ Xa|\ a \qquad (2)$$
$$Y \to Xa|\ a \qquad (3)$$
$$Z \to XY|\ Zb|\ bb|\ aXY|\ Xa|\ a \qquad (4).$$

Lect. univ. dr. TATIANA-CORINA DOSESCU
Facultatea de Informatică
Universitatea Titu Maiorescu