

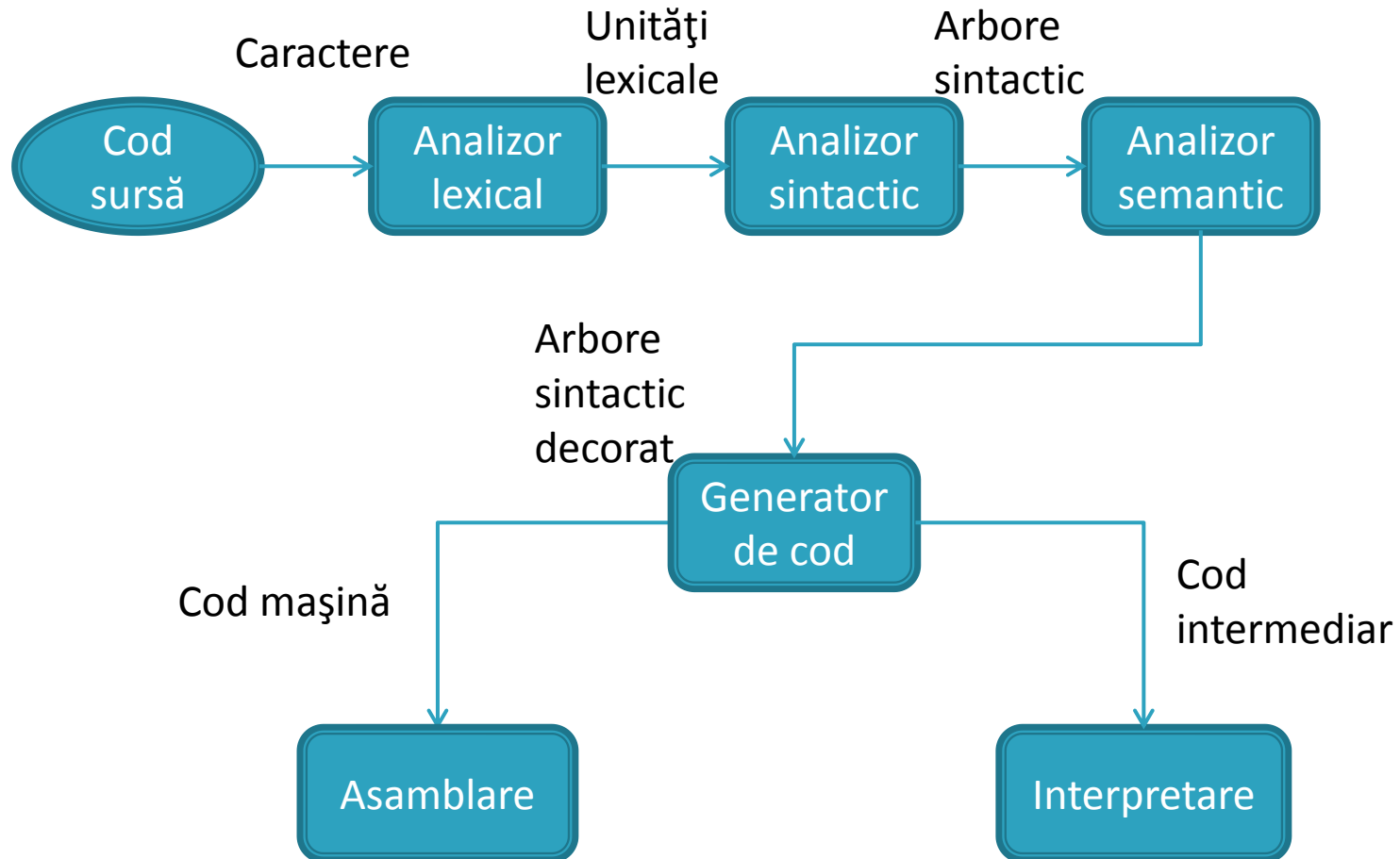
Limbaje formale, automate și compilatoare

Curs 8

Recapitulare

- ▶ Pașii compilării
- ▶ Analiza lexicală
 - Descriere lexicală
 - Interpretare
 - Interpretare orientată dreapta
 - Descriere lexicală bine formată
- ▶ Lex
- ▶ Gramatici de tipul 2

Compilare



Exemplu de analizor lexical

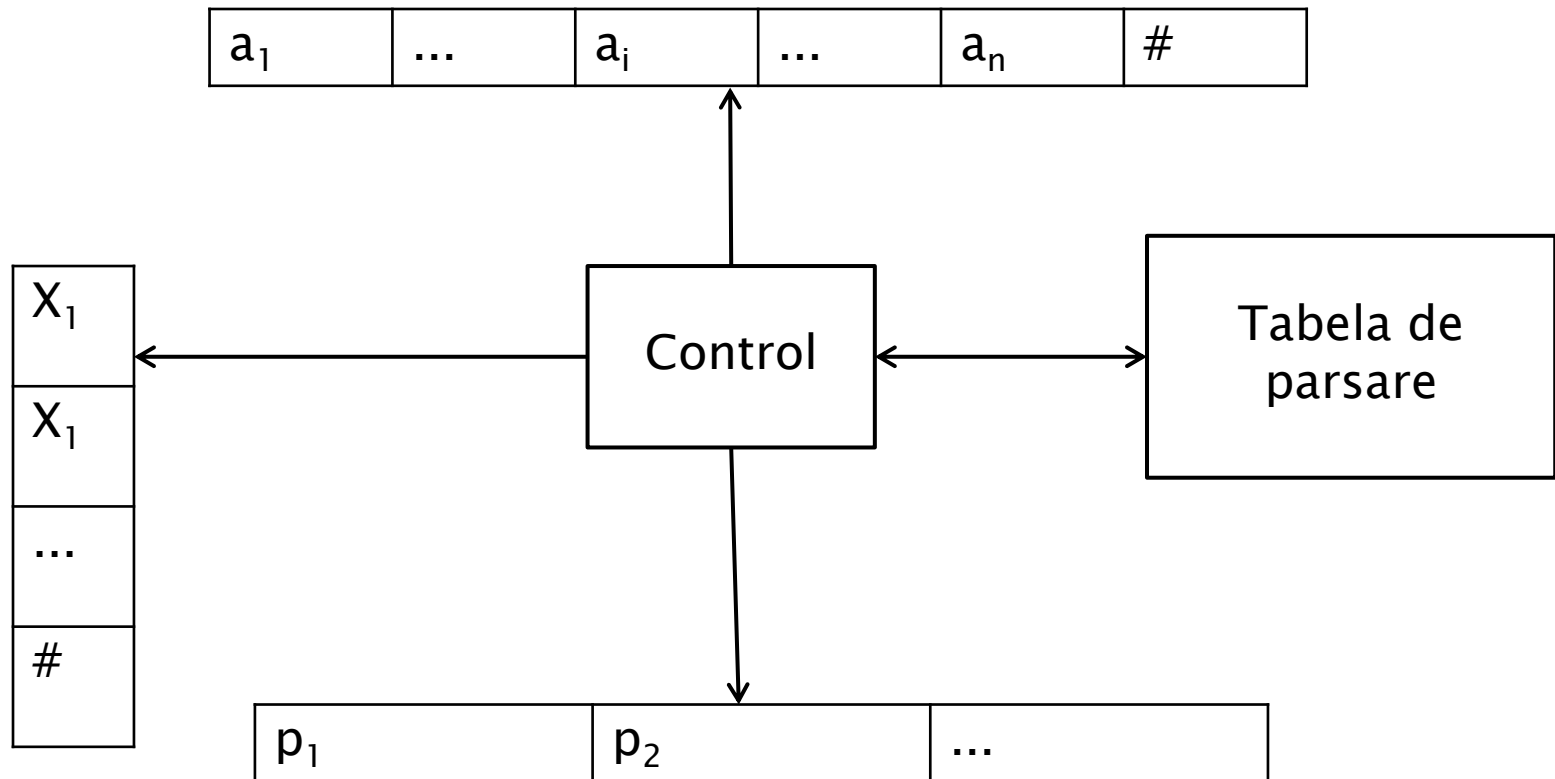
► Fie descrierea lexicală:

- litera $\rightarrow a \mid b \mid \dots \mid z$
- cifra $\rightarrow 0 \mid 1 \mid \dots \mid 9$
- identificator $\rightarrow \text{litera} (\text{litera} \mid \text{cifra})^*$
- semn $\rightarrow + \mid -$
- numar $\rightarrow (\text{semn} \mid \epsilon) \text{cifra}^+$
- operator $\rightarrow + \mid - \mid * \mid / \mid < \mid > \mid <= \mid >= \mid < >$
- asignare $\rightarrow :=$
- doua_puncte $\rightarrow :$
- cuvinte_rezervate $\rightarrow \text{if} \mid \text{then} \mid \text{else}$
- paranteze $\rightarrow) \mid ($

Cuprins

- ▶ Analiza sintactică ascendentă
 - Parser ascendent general
- ▶ Gramatici LR(k)
 - Definiție
 - Proprietăți
- ▶ Gramatici LR(0)
 - Teorema de caracterizare LR(0)
 - Automatul LR(0)

Parser ascendente general



Configurații

- ▶ O configurație $(\# \gamma, u\#, \pi)$ este interpretată în felul următor:
 - $\# \gamma$ este conținutul stivei cu simbolul $\#$ la baza.
 - $u\#$ este conținutul intrării.
 - π este conținutul ieșirii.
- ▶ $C_0 = \{(\#, w\#, \varepsilon) \mid w \in T^*\}$ mulțimea configurațiilor inițiale.

Tranziții

- ▶ Parserul ascendent atașat gramaticii G este perechea (C_0, \vdash) unde C_0 este mulțimea configurațiilor inițiale, iar \vdash este o relație de tranziție definită astfel:
 - $(\# \gamma, au\#, \pi) \vdash (\# \gamma a, u\#, \pi)$ (**deplasare**) pentru orice $\gamma \in \Sigma^*$, $a \in T$, $u \in T^*$, $\pi \in P^*$.
 - $(\# \alpha \beta, u\#, \pi) \vdash (\# \alpha A, u\#, \pi r)$ dacă $r = A \rightarrow \beta$ (**reducere**).
 - Configurația $(\# S, \#, \pi)$ unde $\pi \neq \varepsilon$, se numește **configurație de acceptare**.
 - Orice configurație, diferită de cea de acceptare, care nu este în relația \vdash cu nici o altă configurație este o configurație eroare.
- ▶ Parsere de deplasare/reducere.

Exemplu

- ▶ Fie gramatica $S \rightarrow aSb \mid \varepsilon$. Tranzițiile sunt:
 - $(\# \gamma, u\#, \pi) \vdash (\# \gamma S, u\#, \pi 2)$
 - $(\# \gamma aSb, u\#, \pi) \vdash (\# \gamma S, u\#, \pi 1)$
 - $(\# \gamma, au\#, \pi) \vdash (\# \gamma a, u\#, \pi)$
 - $(\# \gamma, bu\#, \pi) \vdash (\# \gamma b, u\#, \pi)$
- ▶ O succesiune de tranziții se numește calcul
 - $(\#, \#, \varepsilon) \vdash (\# S, \#, 2)$
 - $(\#, aabb\#, \varepsilon) \vdash (\# a, abb\#, \varepsilon) \vdash (\# aa, bb\#, \varepsilon) \vdash (\# aaS, bb\#, 2) \vdash (\# aaSb, b\#, 2) \vdash (\# aS, b\#, 21) \vdash (\# aSb, \#, 21) \vdash (\# S, \#, 211)$

Conflicte

- ▶ Parserul este nedeterminist:
 - Pentru o configurație de tipul $(\# \alpha \beta, au\#, \pi)$, $S \rightarrow \beta$, există două posibilități (conflict **deplasare/reducere**):
 - $(\# \alpha \beta, au\#, \pi) \vdash (\# \alpha S, au\#, \pi r)$ (reducere cu $S \rightarrow \beta$)
 - $(\# \alpha \beta, au\#, \pi) \vdash (\# \alpha \beta a, u\#, \pi)$ (deplasare)
 - Pentru o configurație $(\# \gamma, u\#, \pi)$ cu $\gamma = \alpha_1 \beta_1 = \alpha_2 \beta_2$ și $A \rightarrow \beta_1, B \rightarrow \beta_2$, reguli (conflict **reducere/reducere**)
 - $(\# \alpha_1 \beta_1, u\#, \pi) \vdash (\# \alpha_1 A, au\#, \pi r_1)$
 - $(\# \alpha_2 \beta_2, u\#, \pi) \vdash (\# \alpha_2 B, au\#, \pi r_2)$

Corectitudine

- ▶ Spunem că un cuvânt $w \in T^*$ este acceptat de un parser ascendent dacă există măcar un calcul de forma
 - $(\#, w\#, \varepsilon) \vdash^+(\#S, \#, \pi)$
- ▶ Pentru ca parserul descris să fie corect, trebuie ca el să accepte toate cuvintele din $L(G)$ și numai pe acestea.
- ▶ **Teorema**
 - Parserul ascendent general atașat unei gramatici G este corect: pentru orice $w \in T^*$, $w \in L(G)$ dacă și numai dacă în parser are loc calculul $(\#, w\#, \varepsilon) \vdash^+(\#S, \#, \pi)$.

Analiza sintactică LR

- ▶ Gramatici LR(k): Left to right scanning of the input, constructing a Rightmost derivation in reverse, using k symbols lookahead
- ▶ Definiție
 - O gramatică G se numește gramatică LR(k), $k \geq 0$, dacă pentru orice două derivări de forma:
 - $S' \Rightarrow S \xRightarrow{dr} \alpha A u \xRightarrow{dr} \alpha \beta u = \delta u$
 - $S' \Rightarrow S \xRightarrow{dr} \alpha' A' u' \xRightarrow{dr} \alpha' \beta' u' = \alpha \beta v = \delta v$
 - pentru care $k:u = k:v$, are loc $\alpha = \alpha'$, $\beta = \beta'$, $A = A'$

Analiza sintactică LR

▶ Teorema 1

- Dacă G este gramatică $LR(k)$, $k \geq 0$, atunci G este neambiguă.

▶ Un limbaj L este (în clasa) $\mathcal{LR}(k)$ dacă există o gramatică $LR(k)$ care îl generează

▶ Teorema 2

- Orice limbaj $\mathcal{LR}(k)$ este limbaj de tip 2 determinist.

▶ Teorema 3

- Orice limbaj de tip 2 determinist este limbaj $LR(1)$.

▶ Teorema 4

- Pentru orice limbaj $\mathcal{LR}(k)$, $k \geq 1$, există o gramatică $LR(1)$ care generează acest limbaj, adică $LR(0) \subset LR(1) = LR(k)$, $k \geq 1$.

Gramatici LR(0)

► Definiție

- Fie $G = (V, T, S, P)$ o gramatică independentă de context redusă. Să presupunem că simbolul \bullet nu este în Σ . Un **articol** pentru gramatica G este o producție $A \rightarrow \gamma$ în care s-a adăugat simbolul \bullet într-o anumite poziție din γ . Notăm un articol prin $A \rightarrow \alpha \bullet \beta$ dacă $\gamma = \alpha \beta$. Un articol în care \bullet este pe ultima poziție se numește **articol complet**.

► Definiție

- Un **prefix viabil** pentru gramatica G este orice prefix al unui cuvânt $\alpha\beta$ dacă $S \xRightarrow{dr}^* \alpha A u \xRightarrow{dr} \alpha \beta u$. Dacă $\beta = \beta_1 \beta_2$ și $\varphi = \alpha \beta_1$ spunem că articolul $A \rightarrow \beta_1 \bullet \beta_2$ este **valid** pentru **prefixul viabil** φ .

Exemplu

- ▶ Exemplu $S \rightarrow A, A \rightarrow aAa \mid bAb \mid c \mid \varepsilon$.
 - Articole: $S \rightarrow \bullet A, S \rightarrow A \bullet, A \rightarrow \bullet aAa, A \rightarrow a \bullet Aa, A \rightarrow aA \bullet a, A \rightarrow aAa \bullet, A \rightarrow \bullet bAb, A \rightarrow b \bullet Ab, A \rightarrow bA \bullet b, A \rightarrow bAb \bullet, A \rightarrow \bullet c, A \rightarrow c \bullet, A \rightarrow \bullet$.
- ▶ Articole valide pentru prefixe viabile:

Prefixul viabil	Articole valide	Derivarea corespunzătoare
ab	$A \rightarrow b \bullet Ab$ $A \rightarrow \bullet aAa$ $A \rightarrow \bullet bAb$	$S \Rightarrow A \Rightarrow aAa \Rightarrow abAba$ $S \Rightarrow A \Rightarrow aAa \Rightarrow abAba \Rightarrow abaAaba$ $S \Rightarrow A \Rightarrow aAa \Rightarrow abAba \Rightarrow abbAbba$
ε	$S \rightarrow \bullet A$ $A \rightarrow \bullet bAb$ $A \rightarrow \bullet c$	$S \Rightarrow A$ $S \Rightarrow A \Rightarrow bAb$ $S \Rightarrow A \Rightarrow c$

Gramatici LR(0)

► Lema

- Fie G o gramatică și $A \rightarrow \beta_1 \bullet B \beta_2$ un articol valid pentru prefixul viabil γ . Atunci, oricare ar fi producția $B \rightarrow \beta$, articolul $B \rightarrow \bullet \beta$ este valid pentru γ .

► Teorema (caracterizare LR(0))

- Gramatica G este gramatică LR(0) dacă și numai dacă, oricare ar fi prefixul viabil γ , sunt îndeplinite condițiile:
 - 1. nu există două articole complete valide pentru γ .
 - 2. dacă articolul $A \rightarrow \beta \bullet$ este valid pentru γ , nu există nici un articol $B \rightarrow \beta_1 \bullet a \beta_2$, $a \in T$, valid pentru γ .

Gramatici LR(0)

► Teorema

- Fie $G = (V, T, S, P)$ o gramatică independentă de context. Mulțimea prefixelor viabile pentru gramatica G este limbaj regulat.

► Demonstrație

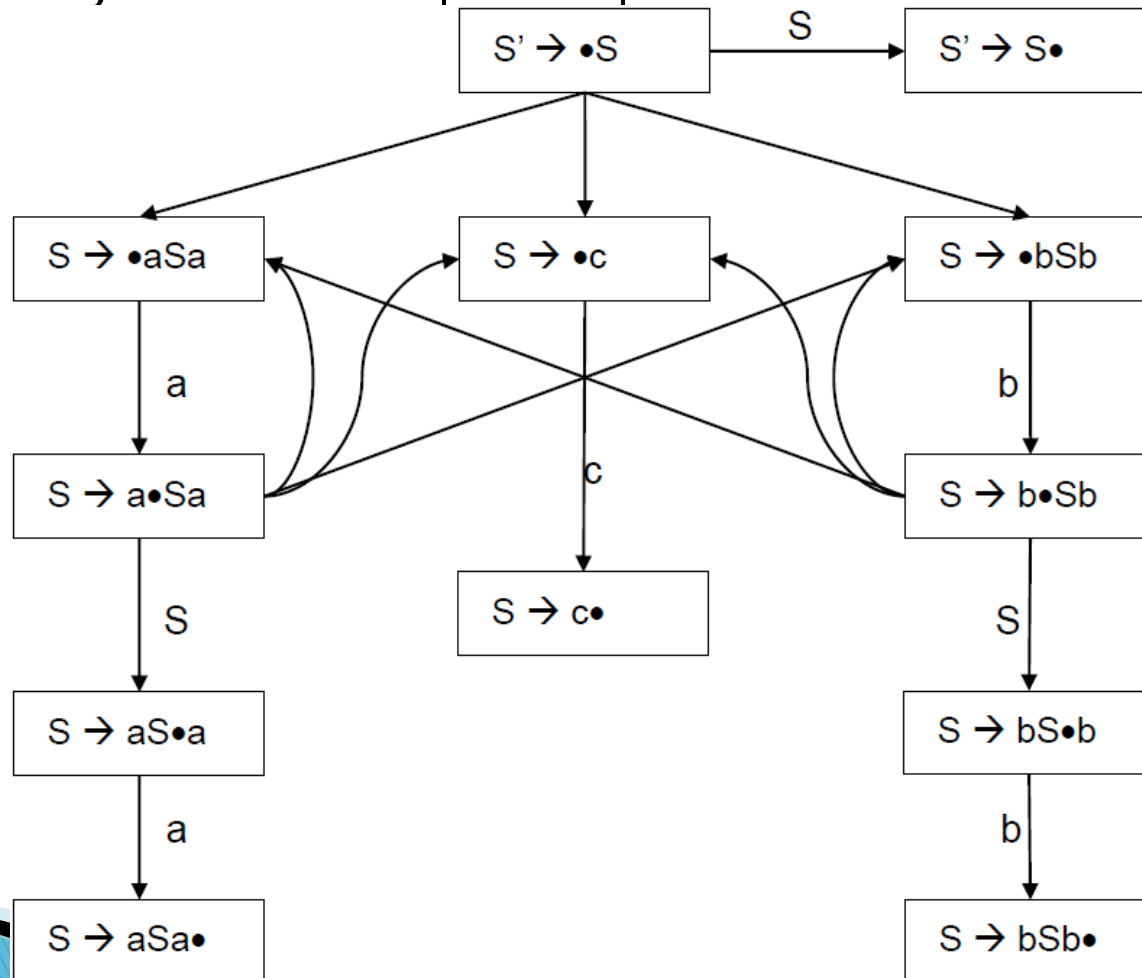
- G' este G la care se adaugă $S' \rightarrow S$.
- $M = (Q, \Sigma, \delta, q_0, Q)$, unde:
 - Q este mulțimea articolelor gramaticii G' ,
 - $\Sigma = V \cup T$, $q_0 = S' \rightarrow \bullet S$
 - $\delta: Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q$ definită astfel:
 - $\delta(A \rightarrow \alpha \bullet B \beta, \epsilon) = \{B \rightarrow \bullet \alpha \mid B \rightarrow \gamma \in P\}$.
 - $\delta(A \rightarrow \alpha \bullet X \beta, X) = \{A \rightarrow \alpha X \bullet \beta\}$, $X \in \Sigma$.
 - $\delta(A \rightarrow \alpha \bullet a \beta, \epsilon) = \emptyset$, $\forall a \in T$.
 - $\delta(A \rightarrow \alpha \bullet X \beta, Y) = \emptyset$, $\forall X, Y \in \Sigma$ cu $X \neq Y$.

► Se arată că are loc:

- $(A \rightarrow \alpha \bullet \beta \in \delta^*(q_0, \gamma) \Leftrightarrow \gamma \text{ este prefix viabil și } A \rightarrow \alpha \bullet \beta \text{ este valid pentru } \gamma.$

Exemplu

- ▶ $S' \rightarrow S, S \rightarrow aSa \mid bSb \mid c$



Gramatici LR(0)

- ▶ **Teorema (caracterizare LR(0)). Demonstrație**
 - \Rightarrow G este LR(0) și, prin reducere la absurd 1 sau 2 nu are loc
 - \Leftarrow 1, 2 au loc si prin reducere la absurd, G nu este LR(0): există
 - $S' \Rightarrow S \xrightarrow{dr}^* \alpha A u \xrightarrow{dr} \alpha \beta u = \delta u$
 - $S' \Rightarrow S \xrightarrow{dr}^* \alpha' A' u' \xrightarrow{dr} \alpha' \beta' u' = \alpha \beta v = \delta v$
 - Nu au loc $\alpha = \alpha', \beta = \beta', A = A'$
 - Fără a restrânge generalitatea, presupunem că
 - $|\delta| = |\alpha \beta| \leq |\alpha' \beta'|$

Gramatici LR(0)

- ▶ Cazul 1: $|\alpha'| \leq |\delta|$
 - $\beta' = \beta_1' \beta_2'$, $v = \beta_2' u'$, $\beta_2' \in T^*$. Din (1) avem $A \rightarrow \beta \bullet$ articol valid pentru δ și din (2) avem $A' \rightarrow \beta_1' \bullet \beta_2'$ articol valid pentru δ . Dacă $\beta_2' = \varepsilon$ atunci contrazic (1) iar altfel contrazic (2).

α	β	u
δ		
α'	β'	u'
δ		v

Gramatici LR(0)

► Cazul 2: $|\alpha'| > |\delta|$

- $\alpha' = \delta u_1$, $v = u_1 \beta' u'$ $\in T^*$ ($|u_1| \geq 1$) În derivarea (2), punem în evidență prima formă propozițională care are prefixul δ .
- $S' \Rightarrow S_{dr} \Rightarrow \alpha_1 A_1 u_{1dr} \Rightarrow \alpha_1 \beta_1 u_{1dr} \Rightarrow \alpha_1 \beta'_1 \beta''_1 u_1 = \delta \beta''_1 u_{1dr} \Rightarrow \delta v$
- Avem $A \rightarrow \beta \bullet$ și $A_1 \rightarrow \beta'_1 \bullet \beta''_1$ articole valid pentru δ

α	β	u	
δ			
α'		β'	u'
δ		v	

Automatul LR(0)

- ▶ Algoritmul 1 (procedura închidere(t))
- ▶ Intrare:
 - Gramatica $G = (V, T, S, P)$;
 - Mulțimea t de articole din gramatica G ;
- ▶ Ieșire: $t' = \text{închidere}(t) = \{q \in Q \mid \exists p \in t, q \in \delta(p, \varepsilon)\} = \delta(t, \varepsilon)$

Automatul LR(0)

- ▶ $t' = t$; flag = true;
- ▶ while(flag) {
 - flag = false;
 - for ($A \rightarrow \alpha \bullet B \beta \in t'$) {
 - for ($B \rightarrow \gamma \in P$)
 - if ($B \rightarrow \bullet \gamma \notin t'$) {
 - $t' = t' \cup \{B \rightarrow \bullet \gamma\}$;
 - flag = true;
 - }//endif
 - }//endforB
 - }//endforA
- ▶ }//endwhile
- ▶ return t' ;

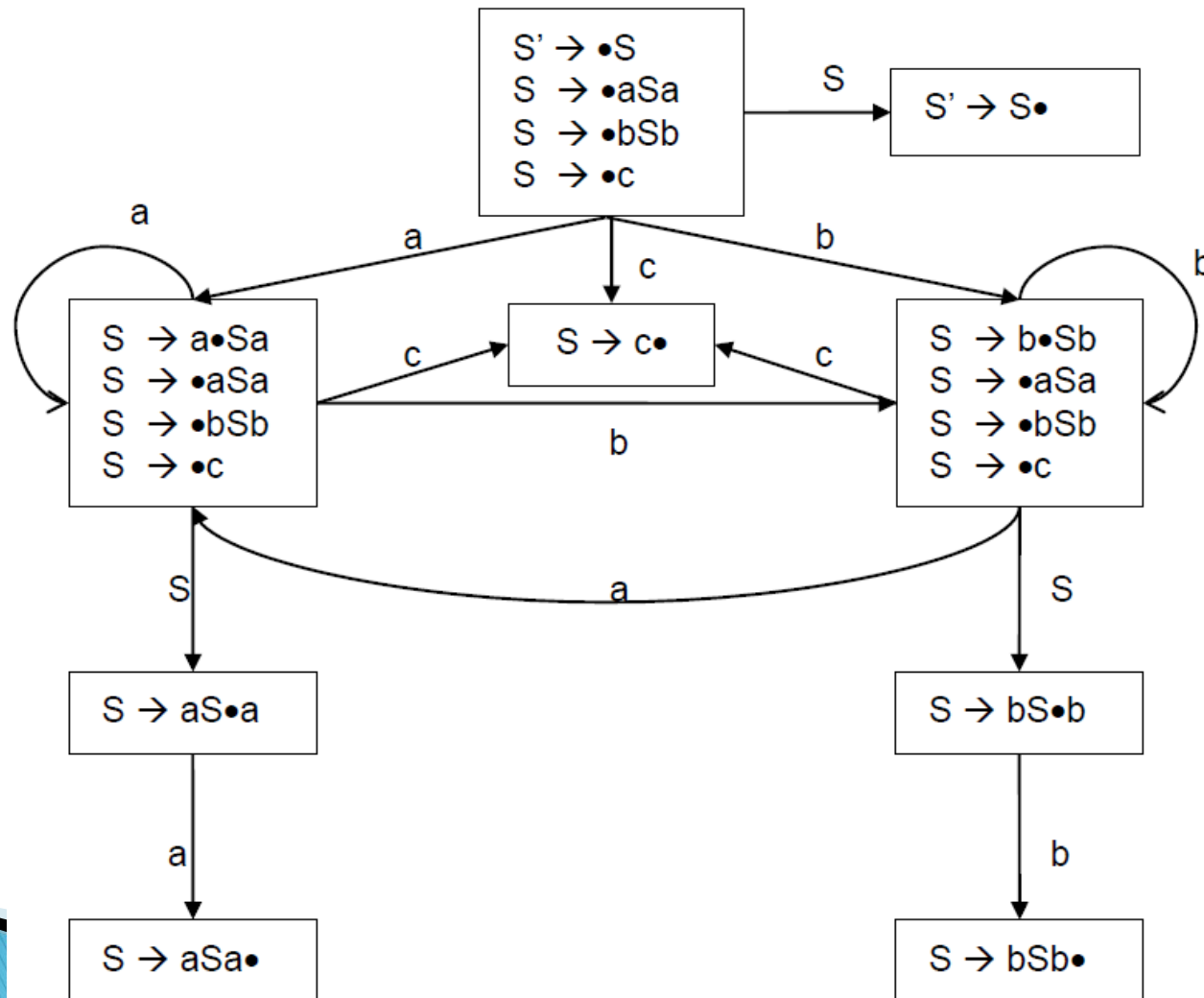
Automatul LR(0)

- ▶ Algoritmul 2 Automatul LR(0)
 - Intrare: Gramatica $G = (N, T, S, P)$ la care s-a adăugat $S' \rightarrow S$;
 - Ieșire: Automatul determinist $M = (T, \Sigma, g, t_0, T)$ echivalent cu M .

- ▶ $t_0 = \text{inchidere}(S' \rightarrow \bullet S); T = \{t_0\}; \text{marcat}(t_0) = \text{false};$
- ▶ $\text{while}(\exists t \in T \ \&\& \ !\text{marcat}(t)) \{ \ // \ \text{marcat}(t) = \text{false}$
 - $\text{for}(X \in \Sigma) \{ \ // \ \Sigma = N \cup T$
 - $t' = \emptyset;$
 - $\text{for}(A \rightarrow \alpha \bullet X \beta \ \epsilon \ T)$
 - $t' = t' \cup \{B \rightarrow \alpha X \bullet \beta \mid A \rightarrow \alpha \bullet X \beta \ \epsilon \ T\};$
 - $\text{if}(t' \neq \emptyset)\{$
 - $t' = \text{inchidere}(t');$
 - $\text{if}(t' \notin T) \{$
 - $T = T \cup \{t'\};$
 - $\text{marcat}(t') = \text{false};$
 - $\} // \text{endif}$
 - $g(t, X) = t';$
 - $\} // \text{endif}$
 - $\} // \text{endfor}$
 - $\} // \text{endfor}$
 - $\text{marcat}(t) = \text{true};$
 - ▶ $\} // \text{endwhile}$

Automatul LR(0) – Exemplu

- ▶ $S' \rightarrow S, S \rightarrow aSa \mid bSb \mid c$

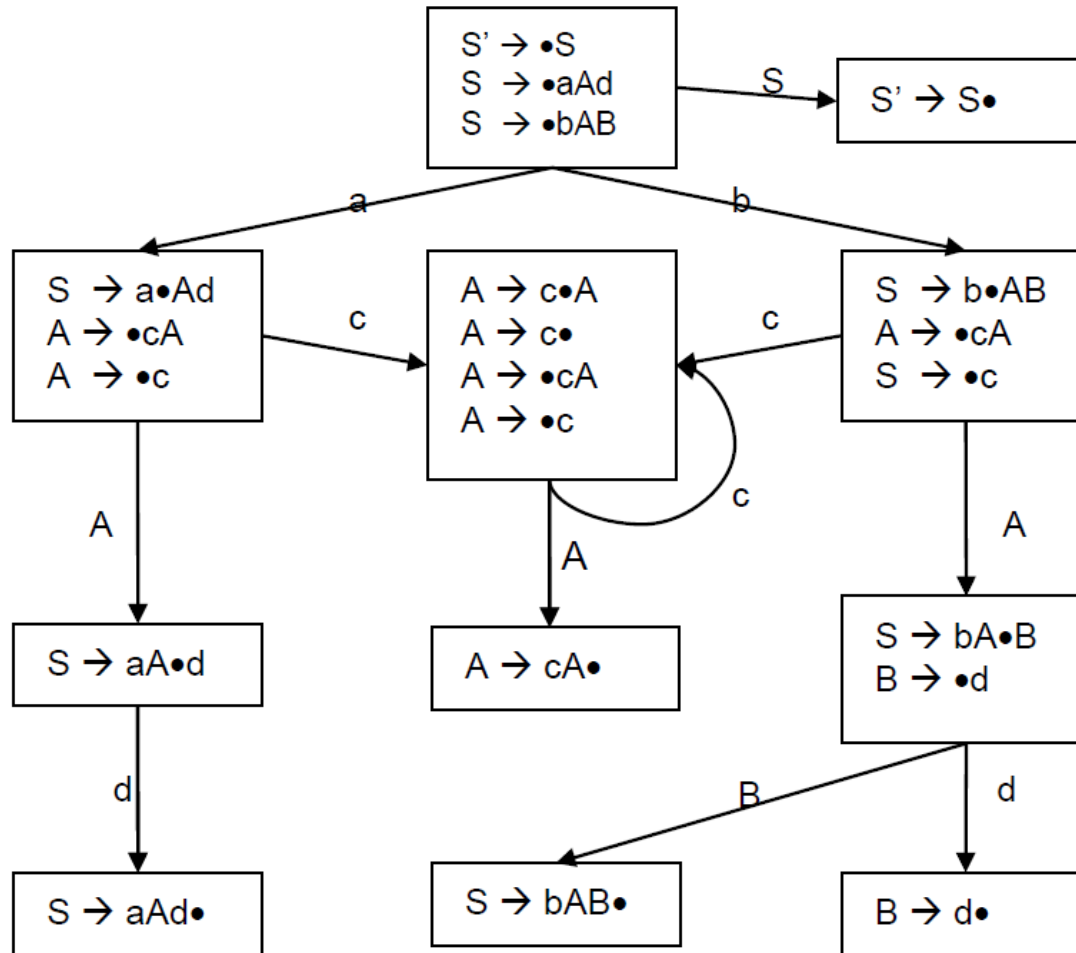


Test LR(0)

- ▶ **Definiție** Fie G o gramatică și M automatul LR(0) atașat lui G .
 - Spunem că o stare a lui M are un conflict **reducere/reducere** dacă ea conține două articole complete distincte $A \rightarrow \alpha \bullet$, $B \rightarrow \beta \bullet$.
 - Spunem că o stare a lui M are un conflict **deplasare/reducere** dacă ea conține un articol complet $A \rightarrow \alpha \bullet$ și un articol cu terminal după punct de forma $B \rightarrow \beta \bullet a \gamma$.
 - Spunem că o stare este **consistentă** dacă ea nu conține conflicte și este **inconsistentă** în caz contrar.
- ▶ **Teorema** Fie G o gramatică și M automatul său LR(0). Gramatica G este LR(0) dacă și numai dacă automatul M nu conține stări inconsistente

Exemplu

- $S \rightarrow aAd \mid bAB$, $A \rightarrow cA \mid c$, $B \rightarrow d$



Bibliografie

- ▶ Grigoraș Gh., *Construcția compilatoarelor. Algoritmi fundamentali*, Editura Universității “Alexandru Ioan Cuza”, Iași, 2005