Fundamentele Informatica 1 (I&E)

najaar 2015

`http://www.liacs.leidenuniv.nl/~vlietrvan1/fi1ie/`

Rudy van Vliet
kamer 124 Snellius, tel. 071-527 5777
rvvliet(at)liacs(dot)nl

college 7, 17 november 2015

4. Context-Free Languages
4.3. Regular Languages and Regular Grammars
4.4. Derivation Trees
4.5. Simplified Forms and Normal Forms

Example 4.1. The language *AnBn*

$AnBn = \{a^i b^i \mid i \geq 0\}$

$S \to aSb \mid \Lambda$

4.3. Regular Languages and Regular Grammars

reg. languages	FA	reg. grammar	reg. expression
determ. cf. languages	DPDA		
cf. languages	PDA	cf. grammar	
re. languages	TM	unrestr. grammar	

A slide from lecture 6:

Theorem 4.9.

If L_1 and L_2 are context-free languages over an alphabet Σ, then

$$L_1 \cup L_2, \quad L_1 L_2 \quad \text{and } L_1^*$$

are also CFLs.

Proof. . .

A slide from lecture 4:

Definition 3.1. Regular Languages over an Alphabet Σ.

If Σ is an alphabet,
the set $\mathcal{R}$ of regular languages over Σ is defined as follows.

1. The language $\emptyset$ is an element of $\mathcal{R}$,
   and for every $\sigma \in \Sigma$, the language $\{\sigma\}$ is in $\mathcal{R}$.

2. For any two languages L_1 and L_2 in $\mathcal{R}$,
   the three languages
   $L_1 \cup L_2$, $L_1 L_2$, and L_1^*
   are elements of $\mathcal{R}$.

(and nothing more)

Exercise.

- Give a context-free grammar G_1, such that $L(G_1) = \emptyset$.

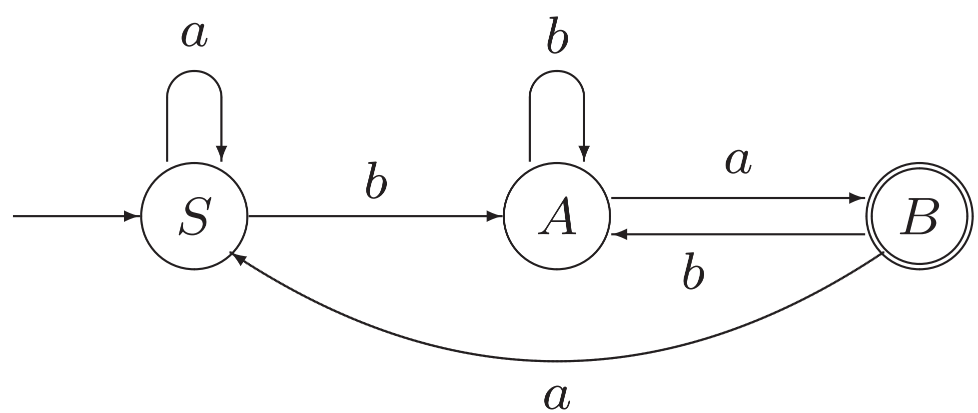
- Let $\sigma \in \Sigma$.
  Give a context-free grammar G_2, such that $L(G_2) = \{\sigma\}$.

Example 4.11. A CFG Corresponding to a Regular Expression.

$$bba(ab)^* + (ab + ba^*b)^*ba$$

In general: construction of a CFG from a finite automaton.

Example: an FA accepting $\{a, b\}^*\{ba\}$



Definition 4.13. Regular Grammars.

A context-free grammar $G = (V, \Sigma, S, P)$ is *regular* if every production is of the form

$$A \to \sigma B \quad \text{or} \quad A \to \Lambda,$$

where $A, B \in V$ and $\sigma \in \Sigma$.

Theorem 4.14.

For every language $L \subseteq \Sigma^*$,
L is regular,
if and only if $L = L(G)$ for some regular grammar G.

Hence, the term ‘regular grammar’ is appropriate.

Proof. . .

Example.

An FA corresponding to regular grammar with productions

$$S \to aA \mid bC \qquad A \to aS \mid bC \qquad C \to aA \mid bS \mid \Lambda$$

Example.

An NFA corresponding to regular grammar with productions

$$S \to aA \mid bC \quad A \to aS \mid bC \quad C \to aA \mid aS \mid \Lambda$$

Exercise.

Let $G = (V, \Sigma, S, P)$ be an arbitrary regular grammar.

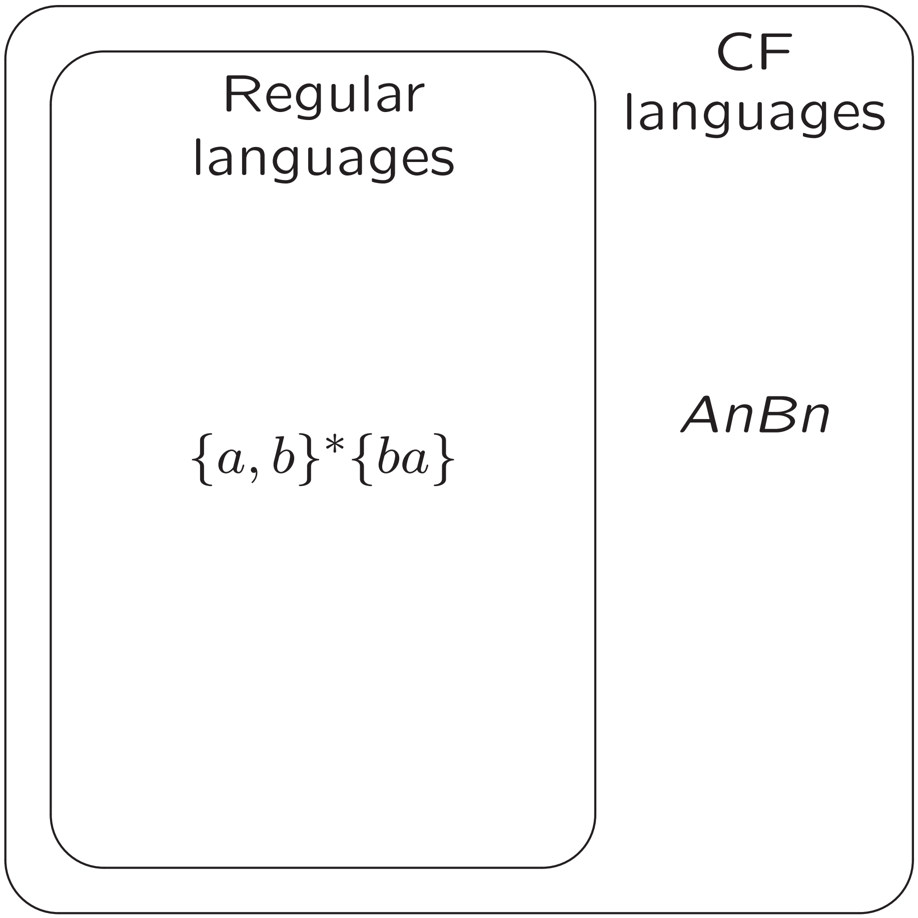
Specify an NFA $M = (Q, \Sigma, q_0, A, \delta)$, such that $L(M) = L(G)$.

Can we find regular grammar for:

Example 4.1. The language *AnBn*

$AnBn = \{a^i b^i \mid i \geq 0\}$

$S \to aSb \mid \Lambda$


CF
languages
Regular
languages
$\{a, b\}^*\{ba\}$
AnBn

4.4. Derivation Trees

Exercise.

a. Give a derivation of $aaabbbbb$ in the following grammar G with start variable S:

$$S \to TB \quad T \to aTb \mid ab \quad B \to bB \mid \Lambda$$

b. Give a derivation of $a+a+a$ in the following grammar G with start variable S:

$$S \to a \mid S+S \mid S*S \mid (S)$$

Useful to consider *how* a string is generated.

Visualize this by means of a tree.

From derivation to derivation tree:

Root node $\approx$ start variable S

Each step in derivation corresponds to application of production $A \to \alpha$ to some occurrence of A.

In tree: give corresponding node labelled by A children labelled by symbols of α (in right order).

If $\alpha = \Lambda$ . . .

Yield of tree. . .

For each derivation in a CFG, there is exactly one derivation tree

Example 4.2. The language *Expr*

$$S \to a \mid S + S \mid S * S \mid (S)$$

$$\underline{S} \Rightarrow \underline{S}{+}S \Rightarrow a{+}\underline{S} \Rightarrow a{+}(\underline{S}) \Rightarrow a{+}(\underline{S}{*}S) \Rightarrow a{+}(a{*}\underline{S}) \Rightarrow a{+}(a{*}a)$$

Conversely, . . .

Definition 4.16. Leftmost and Rightmost Derivations

A derivation in a context-free grammar is a *leftmost* derivation (LMD)
if, at each step, a production is applied to the leftmost variable-occurrence in the current string.
A rightmost derivation (RMD) is defined similarly.

Exercise.

Let G be the following grammar with start variable S:

$$S \to TB \quad T \to aTb \mid ab \quad B \to bB \mid \Lambda$$

Construct the derivation tree of $aaabbbbb$ in G corresponding to the following derivation:

$$\begin{aligned} \underline{S} &\Rightarrow T\underline{B} \Rightarrow \underline{T}bB \Rightarrow a\underline{T}bbB \Rightarrow aaTbbb\underline{B} \\ &\Rightarrow aaTbbbb\underline{B} \Rightarrow aa\underline{T}bbbb \Rightarrow aaabbbbb \end{aligned}$$

Top-down vs. bottom-up construction derivation tree. . .

4.5. Simplified Forms and Normal Forms

Given string x and CFG G, is x generated by G?

Try all possible derivations: 1 step, 2 steps, ...

Let γ be current string,
let $|\gamma|$ be length of γ (as usual),
let t_γ be number of terminals in γ

Example 4.2. The language *Expr*

$$S \to a \mid S + S \mid S * S \mid (S)$$

$$\underline{S} \Rightarrow \underline{S}+S \Rightarrow a+\underline{S} \Rightarrow a+(\underline{S}) \Rightarrow a+(\underline{S}*S) \Rightarrow a+(a*\underline{S}) \Rightarrow a+(a*a)$$

Example 4.2. The language *Expr*

$$S \rightarrow a \mid S + S \mid S * S \mid (S)$$

γ	$\underline{S}$	$\underline{S} + S$	$a + \underline{S}$	$a + (\underline{S})$	$a + (\underline{S} * S)$	$a + (a * \underline{S})$	$a + (a * a)$
$\lvert\gamma\rvert$	1	3	3	5	7	7	7
t_γ	0	1	2	4	5	6	7
$\lvert\gamma\rvert + t_\gamma$	1	4	5	9	12	13	14

If G has no Λ-productions,...

If G has no unit-productions $A \to B$, either,...

Definition 4.29. Chomsky Normal Form

A context-free grammar is said to be in *Chomsky normal form* if every production is of one of these two types:

$$A \rightarrow BC \text{ (where } B \text{ and } C \text{ are variables)}$$
$$A \rightarrow \sigma \text{ (were } \sigma \text{ is a terminal symbol)}$$

Arbitrary CFG may have

- productions $A \to \Lambda$
- productions $A \to B$ (unit productions)
- productions $A \to bc$, $A \to Bc$, $A \to bC$
- productions $A \to \alpha$ with $|\alpha| \geq 3$

Converting a CFG to Chomsky Normal Form
Step 1

Removing Λ-productions

Example.

$S \to aSb \mid aBb \quad B \to bB \mid \Lambda$

Converting a CFG to Chomsky Normal Form
Step 1

- Identify *nullable* variables
- Add productions in which nullable variables are removed from right hand side
- Delete Λ-productions
- Delete productions $A \to A$

We cannot generate Λ anymore

Example.

$S \to aSb \mid aBb \quad B \to bB \mid \Lambda$

$S \to SaS \mid B \quad B \to bB \mid \Lambda$

Arbitrary CFG may have

- productions $A \to \Lambda$
- productions $A \to B$ (unit productions)
- productions $A \to bc$, $A \to Bc$, $A \to bC$
- productions $A \to \alpha$ with $|\alpha| \geq 3$

Converting a CFG to Chomsky Normal Form Step 2

- Identify A-*derivable* variables

- For every A-derivable variable B and nonunit production $B \to \alpha$, add production $A \to \alpha$

- Delete unit productions

Example.

$S \to aSb \mid B \quad B \to bB \mid b \mid A \quad A \to aBS \mid a$