Theory of Machines and Languages

Fatemeh Deldar

1403-1404

Methods for Transforming Grammars

- **Substitution rule**
  - **Example**

$$A \rightarrow a\,|aaA|\,abBc,$$
$$B \rightarrow abbA|b.$$

$$A \rightarrow a\,|aaA|\,ababbAc|abbc,$$
$$B \rightarrow abbA|b.$$

- **Removing useless productions**
  - **Remove productions from a grammar that can never take part in any derivation**
  - **Example**
    - **In the following grammar, the production $S \rightarrow A$ clearly plays no role, as A cannot be transformed into a terminal string**

$$S \rightarrow aSb\,|\lambda|\,A,$$
$$A \rightarrow aA,$$

Methods for Transforming Grammars

- **Removing useless productions**
  - **Example**
    - **In the following grammar, the variable *B* is useless**

$$S \to A,$$
$$A \to aA|\lambda,$$
$$B \to bA,$$

  - **Two reasons why a variable is useless:**
    1. ***It cannot be reached from the start symbol***
    2. ***It cannot derive a terminal string***

  - **Example**

$$S \to aS\,|A|\,C,$$
$$A \to a,$$
$$B \to aa,$$
$$C \to aCb.$$

$$S \to aS|A,$$
$$A \to a.$$

Removing λ-Productions

- Any production of a context-free grammar of the form

$$A \to \lambda$$

is called a **λ-production**. Any variable A for which the derivation

$$A \overset{*}{\Rightarrow} \lambda$$

is possible is called **nullable**.

- **Example**

$$\begin{aligned} S &\to aS_1b, \\ S_1 &\to aS_1b|\lambda, \end{aligned} \quad \Rightarrow \quad \begin{aligned} S &\to aS_1b|ab, \\ S_1 &\to aS_1b|ab. \end{aligned}$$

Removing λ-Productions

- **Let G be any context-free grammar with λ not in $L(G)$. Then there exists an equivalent grammar $\hat{G}$ having no λ-productions**

- **Example**

$$
\begin{aligned}
S &\to ABaC,\\
A &\to BC,\\
B &\to b|\lambda,\\
C &\to D|\lambda,\\
D &\to d.
\end{aligned}
\quad\Rightarrow\quad
\begin{aligned}
S &\to ABaC\,|BaC|\,AaC\,|ABa|\,aC\,|Aa|\,Ba|a,\\
A &\to B\,|C|\,BC,\\
B &\to b,\\
C &\to D,\\
D &\to d.
\end{aligned}
$$

Removing Unit-Productions

- Any production of a context-free grammar of the form

$$A \to B,$$

where $A, B \in V$, is called a **unit-production**.

- **To remove unit-productions, we use the substitution rule**

Removing Unit-Productions

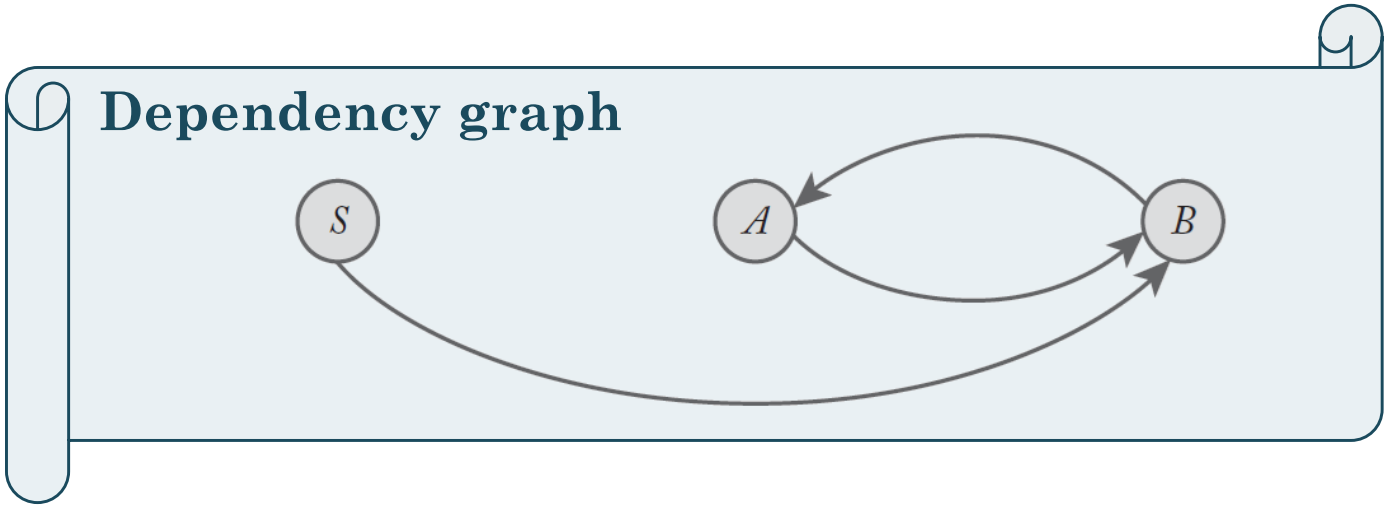
- **Example**

$$\begin{aligned} S &\to Aa|B, \\ B &\to A|bb, \\ A &\to a\,|bc|\,B. \end{aligned}$$

$$\begin{aligned} S &\to Aa, \\ A &\to a|bc, \\ B &\to bb, \end{aligned}$$

$$\begin{aligned} S &\to a\,|bc|\,bb, \\ A &\to bb, \\ B &\to a|bc, \end{aligned}$$

$$\begin{aligned} S &\to a\,|bc|\,bb|Aa, \\ A &\to a\,|bb|\,bc, \\ B &\to a\,|bb|\,bc. \end{aligned}$$



Removing Unit-Productions

- **Let L be a context-free language that does not contain λ. Then there exists a context-free grammar that generates L and that does not have any useless productions, λ-productions, or unit-productions**
  - **We can remove all undesirable productions using the following sequence of steps:**
    1. **Remove λ-productions**
    2. **Remove unit-productions**
    3. **Remove useless productions**

Chomsky Normal Form

- A context-free grammar is in Chomsky normal form if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

- **Example**

$$S \to AS|a,$$
$$A \to SA|b$$

$$S \to AS|AAS,$$
$$A \to SA|aa$$

Chomsky Normal Form

- **Example** Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

$$S \to AD_1,$$
$$D_1 \to BB_a,$$
$$A \to B_aD_2,$$
$$D_2 \to B_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$