Simplifications of Context-Free Grammars

A Substitution Rule

$$S \to aB$$

$$A \to aaA$$

$$A \to abBc$$

$$B \to aA$$

$$B \to b$$

Substitute

$$B \to b$$

Equivalent grammar

$$S \to aB \mid ab$$

$$A \to aaA$$

$$A \to abBc \mid abbc$$

$$B \to aA$$

$$S \to aB \mid ab$$

$$A \to aaA$$

$$A \to abBc \mid abbc$$

$$B \to aA$$

Substitute

$$B \to aA$$

$$S \to \cancel{aB} \mid ab \mid aaA$$

$$A \to aaA$$

$$A \to \cancel{abBc} \mid abbc \mid abaAc$$

Equivalent grammar

In general: $A \to xBz$

$$B \to y_1$$

Substitute $B \to y_1$

$$A \to xBz \mid xy_1z$$

equivalent grammar

Nullable Variables

$\varepsilon-$production: $X \rightarrow \varepsilon$

Nullable Variable: $Y \Rightarrow \mathrm{K} \Rightarrow \varepsilon$

Example:

$$S \rightarrow aMb$$

$$M \rightarrow aMb$$

$$M \rightarrow \varepsilon$$

Nullable variable

$\varepsilon-$production

Removing $\varepsilon-\text{productions}$

$$S \to aMb$$

$$M \to aMb$$

$$\cancel{M \to \varepsilon}$$

Substitute

$$M \to \varepsilon$$

$$S \to aMb \mid ab$$

$$M \to aMb \mid ab$$

After we remove all the $\varepsilon-\text{productions}$ all the nullable variables disappear (except for the start variable)

Unit-Productions

Unit Production: $X \to Y$

(a single variable in both sides)

Example:

$$S \to aA$$

$$A \to a$$

$$\boxed{A \to B}$$

$$\boxed{B \to A}$$

$$B \to bb$$

Unit Productions

Removal of unit productions:

$S \to aA$

$A \to a$

$\xcancel{A \to B}$

$B \to A$

$B \to bb$

Substitute

$A \to B$

$S \to aA \mid aB$

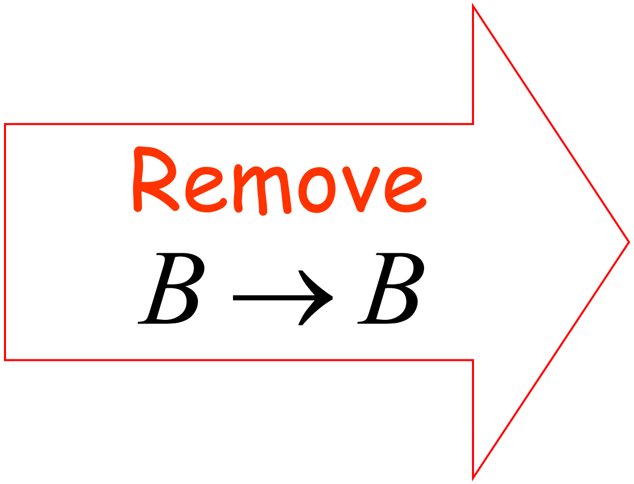
$A \to a$

$B \to A \mid B$

$B \to bb$

Unit productions of form $X \to X$ can be removed immediately

$$S \to aA \mid aB$$
$$A \to a$$
$$B \to A \mid \cancel{B}$$
$$B \to bb$$



Remove

$$B \to B$$

$$S \to aA \mid aB$$
$$A \to a$$
$$B \to A$$
$$B \to bb$$

$$S \to aA \mid aB$$

$$A \to a$$

$$\cancel{B \to A}$$

$$B \to bb$$

Substitute

$$B \to A$$

$$S \to aA \mid aB \mid aA$$

$$A \to a$$

$$B \to bb$$

Remove repeated productions

$$S \to \enclose{circle}{aA} \mid aB \mid \xcancel{aA}$$

$$A \to a$$

$$B \to bb$$

$$\Longrightarrow$$

Final grammar

$$S \to aA \mid aB$$

$$A \to a$$

$$B \to bb$$

Useless Productions

$$S \rightarrow aSb$$

$$S \rightarrow \varepsilon$$

$$S \rightarrow A$$

$$A \rightarrow aA$$

Useless Production

Some derivations never terminate...

$$S \Rightarrow A \Rightarrow aA \Rightarrow aaA \Rightarrow \mathrm{K} \Rightarrow aa\mathrm{K}\, aA \Rightarrow \mathrm{K}$$

Another grammar:

$$S \rightarrow A$$

$$A \rightarrow aA$$

$$A \rightarrow \varepsilon$$

$$\boxed{B \rightarrow bA}$$ Useless Production

Not reachable from S

In general:

If there is a derivation

$$S \Rightarrow \mathrm{K} \Rightarrow xAy \Rightarrow \mathrm{K} \Rightarrow w \quad \in L(G)$$

consists of terminals

Then variable A is useful

Otherwise, variable A is useless

A production $A \to x$ is useless
if any of its variables is useless

	$S \to aSb$	
	$S \to \varepsilon$	Productions
Variables	$S \to A$	useless
useless	$A \to aA$	useless
useless	$B \to C$	useless
useless	$C \to D$	useless

Removing Useless Variables and Productions

Example Grammar:

$$S \to aS \mid A \mid C$$

$$A \to a$$

$$B \to aa$$

$$C \to aCb$$

First: find all variables that can produce strings with only terminals or ε (possible useful variables)

$$S \rightarrow aS \mid A \mid C$$

$$A \rightarrow a$$

$$B \rightarrow aa$$

$$C \rightarrow aCb$$

Round 1: $\{A, B\}$

(the right hand side of production that has only terminals)

Round 2: $\{A, B, S\}$

(the right hand side of a production has terminals and variables of previous round)

This process can be generalized

Then, remove productions that use variables other than $\{A, B, S\}$

$$S \to aS \mid A \mid \cancel{C}$$

$$A \to a$$

$$B \to aa$$

$$\cancel{C \to aCb}$$

$$\Longrightarrow$$

$$S \to aS \mid A$$

$$A \to a$$

$$B \to aa$$

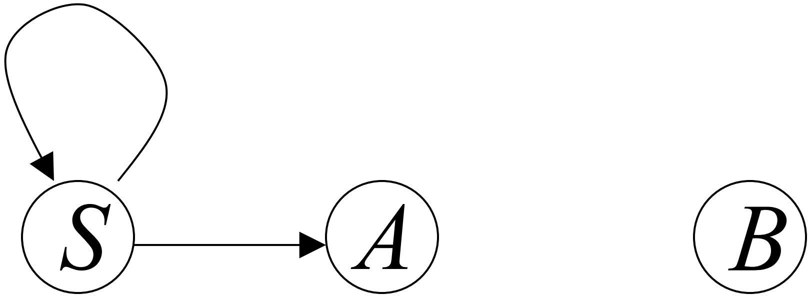
Second: Find all variables reachable from S

Use a Dependency Graph where nodes are variables

$$S \to aS \mid A$$

$$A \to a$$

$$B \to aa$$



unreachable

Keep only the variables reachable from S

$$S \to aS \mid A$$

$$A \to a$$

$$\xcancel{B \to aa}$$

$\Rightarrow$

Final Grammar

$$S \to aS \mid A$$

$$A \to a$$

Contains only useful variables

Removing All

Step 1: Remove Nullable Variables

Step 2: Remove Unit-Productions

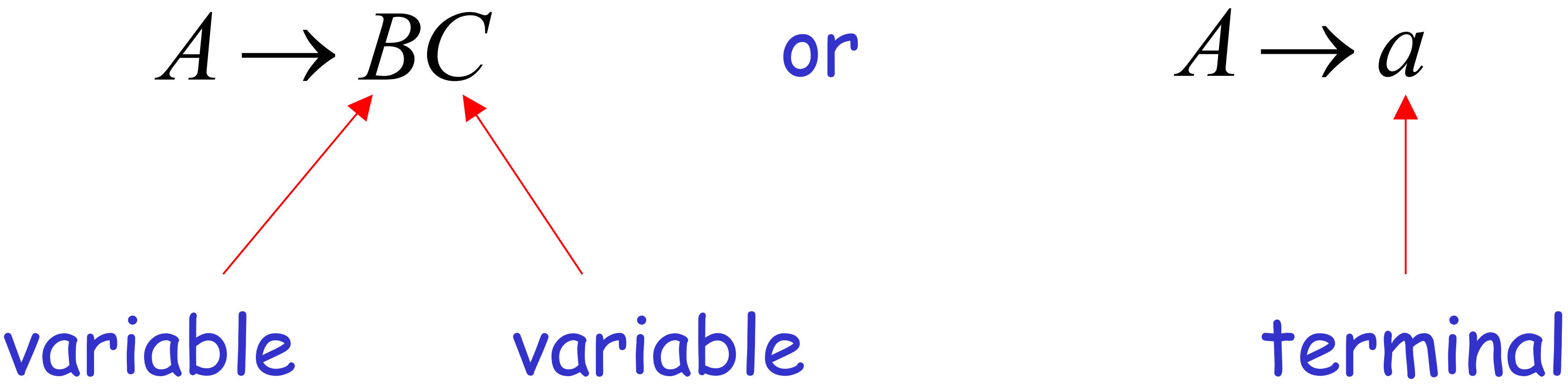
Step 3: Remove Useless Variables

This sequence guarantees that unwanted variables and productions are removed

Normal Forms
for
Context-free Grammars

Chomsky Normal Form

Each production has form:



$A \to BC$ or $A \to a$

variable variable terminal

Examples:

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

Chomsky
Normal Form

$$S \to AS$$
$$S \to \boxed{AAS}$$
$$A \to SA$$
$$A \to \boxed{aa}$$

Not Chomsky
Normal Form

Conversion to Chomsky Normal Form

Example:

$$S \to ABa$$

$$A \to aab$$

$$B \to Ac$$

Not Chomsky Normal Form

We will convert it to Chomsky Normal Form

Introduce new variables for the terminals:

$$T_a, T_b, T_c$$

$$\begin{array}{l} S \to ABa \\ A \to aab \\ B \to Ac \end{array} \quad \Longrightarrow \quad \begin{array}{l} S \to ABT_a \\ A \to T_aT_aT_b \\ B \to AT_c \\ \boxed{\begin{array}{l} T_a \to a \\ T_b \to b \\ T_c \to c \end{array}} \end{array}$$

Introduce new intermediate variable V_1 to break first production:

$$\boxed{S \to ABT_a}$$

$$A \to T_aT_aT_b$$

$$B \to AT_c$$

$$T_a \to a$$

$$T_b \to b$$

$$T_c \to c$$

$$\Longrightarrow$$

$$\boxed{\begin{aligned} S &\to AV_1 \\ V_1 &\to BT_a \end{aligned}}$$

$$A \to T_aT_aT_b$$

$$B \to AT_c$$

$$T_a \to a$$

$$T_b \to b$$

$$T_c \to c$$

Introduce intermediate variable: V_2

$$S \to AV_1$$

$$V_1 \to BT_a$$

$$\boxed{A \to T_aT_aT_b}$$

$$B \to AT_c$$

$$T_a \to a$$

$$T_b \to b$$

$$T_c \to c$$

⇒

$$S \to AV_1$$

$$V_1 \to BT_a$$

$$\boxed{\begin{array}{l} A \to T_aV_2 \\ V_2 \to T_aT_b \end{array}}$$

$$B \to AT_c$$

$$T_a \to a$$

$$T_b \to b$$

$$T_c \to c$$

Final grammar in Chomsky Normal Form:

Initial grammar

$$S \to ABa$$

$$A \to aab$$

$$B \to Ac$$

$$S \to AV_1$$

$$V_1 \to BT_a$$

$$A \to T_aV_2$$

$$V_2 \to T_aT_b$$

$$B \to AT_c$$

$$T_a \to a$$

$$T_b \to b$$

$$T_c \to c$$

In general:

From any context-free grammar
(which doesn't produce ε)
not in Chomsky Normal Form

we can obtain:

an equivalent grammar
in Chomsky Normal Form

The Procedure

First remove:

Nullable variables

Unit productions

(Useless variables optional)

Then, for every symbol a:

New variable: T_a

Add production $T_a \to a$

In productions with length at least 2 replace a with T_a

Productions of form $A \to a$ do not need to change!

Replace any production $A \to C_1C_2\Lambda\ C_n$

with $A \to C_1V_1$

$V_1 \to C_2V_2$

K

$V_{n-2} \to C_{n-1}C_n$

New intermediate variables: $V_1, V_2, \mathrm{K}\ , V_{n-2}$

Observations

- Chomsky normal forms are good for parsing and proving theorems

- It is easy to find the Chomsky normal form for any context-free grammar (which doesn't generate ε)

Greinbach Normal Form

All productions have form:

$$A \to a\, V_1 V_2 \Lambda\, V_k \qquad\qquad k \geq 0$$

symbol

variables

Examples:

$$S \to cAB$$

$$A \to aA \mid bB \mid b$$

$$B \to b$$

Greinbach
Normal Form

$$S \to abSb$$

$$S \to aa$$

Not Greinbach
Normal Form

Conversion to Greinbach Normal Form:

$$S \to abSb$$
$$S \to aa$$

$$\Longrightarrow$$

$$S \to aT_bST_b$$
$$S \to aT_a$$
$$T_a \to a$$
$$T_b \to b$$

Greinbach
Normal Form

Observations

- Greinbach normal forms are very good for parsing strings (better than Chomsky Normal Forms)

- However, it is difficult to find the Greinbach normal of a grammar