Elimination of useless symbols

Given a CFG G we can find an equivalent CFG G_1 such that G_1 has no useless symbols.

We have to first find the set of generating symbols פ.

For this we carry out the procedure.

1) פ <- ∑ since a ->* a for every a in ∑.

2) if X –> X_1 X_2 ..... X_k and X_i are in פ then insert X in פ.

3) Repeat 2) until no more insertions can be made.

פ gives the set of generating symbols. Other symbols can be dropped.

To identify R, the set of reachable symbols in G we carry out the following steps.

1) R = {S}

2) If A is in R and A -> X_1 X_2 …… , X_k insert X_1 , X_2, …… X_k in R.

3) Repeat 2) until no more insertions can be made.

R is the set of all reachable symbols. Other symbols can be dropped.

Example : S -> a A a | a B b

A -> a S | b D

B -> a B a | b

C -> a b b | D D

D -> a D a

Here a, b are generating. Hence B, C are generating. Hence S is generating and therefore A is generating. So D is non-generating.

Getting rid of D we get : S -> a A a | a B b

A -> a S

B -> a B a | b

C -> a b b

S is reachable. Hence a, A, B and b are reachable. C is not reachable.

So getting rid of C we get an equivalent grammar with no useless symbols. S -> a A a | a B b

A -> a S

B -> a B a | b

Note that we have to eliminate the non-generating symbols first. For example if we start with the grammar

S -> A B | a

B -> b

and get rid of the unreachable symbols first we find that all the symbols are reachable. After that if we get rid of the non-generating symbols, A will get dropped and we will be left with the grammar

S -> a

B -> b

which still has B unreachable. However if we get rid of the non-generating symbols first, we get the grammar

S -> a

B -> b

Here B is unreachable. Dropping B we get the grammar

S -> a

which is equivalent to to the original grammar and has no useless symbols.

Points to note :

We saw that we can determine whether ε is generated by a grammar G by checking whether S is nullable.

Similarly we can determine whether L(G) is non-empty by checking whether S is generating.