

## FROM LEFT-REGULAR TO GREIBACH NORMAL FORM GRAMMARS

A. NIJHOLT

*Vrije Universiteit, Department of Mathematics, Amsterdam, The Netherlands*

Received 10 April 1978; revised version received 29 January 1979

Regular grammars, covers, Greibach normal form

### 1. Introduction

Each context-free grammar can be transformed to a context-free grammar in Greibach normal form, that is, a context-free grammar where each right-hand side of a production begins with a terminal symbol and the remainder of the right-hand side consists of nonterminal symbols.

In this short paper we show that for a left-regular grammar  $G$  we can obtain a right-regular grammar  $G'$  (which is by definition in Greibach normal form) which left-to-right covers  $G$  (in this case left parses of  $G'$  can be mapped by a homomorphism on right parses of  $G$ ).

Moreover, it is possible to obtain a context-free grammar  $G''$  in Greibach normal form which right covers the left-regular grammar  $G$  (in this case right parses of  $G''$  are mapped on right parses of  $G$ ).

More general results on covers for context-free grammars in Greibach normal form and in non-left-recursive form can be found in [6,8,13]. Negative cover results appear in [14]. Complexity problems for covers for regular and linear grammars can be found in [4,5]. Covers, in the sense that we will use them here, were introduced by Gray and Harrison [2,3] and they are also considered in [1]. However, as will be seen, our definition is slightly different from theirs, and this difference is essential to obtain the results in this paper. Other papers in which types of covers are introduced are [10,11].

*Preliminaries.* We assume that the reader is familiar with Aho and Ullman [1]. For notational reasons we

review some concepts. A *context-free grammar* is denoted by the 4-tuple  $G = (N, T, P, S)$ , where  $N$  consists of the nonterminals (denoted by the uppercase Latin letters  $A, B, C, \dots$ ),  $T$  consists of the terminals (denoted by the lowercase Latin letters  $a, b, c, \dots$ ),  $P$  is the set of productions,  $P \subset N \times (N \cup T)^*$  (notation  $A \rightarrow \alpha$  for  $(A, \alpha) \in P$ ) and  $S$  is the start symbol. The language generated by a context-free grammar  $G$  is denoted by  $L(G)$ . The productions are assumed to be labeled by the numbers 1 to  $|P|$ , where  $|P|$  denotes the number of elements in  $P$ . Leftmost and rightmost derivations are defined as usual. A *left parse* of a sentence  $w \in L(G)$  is the string of numbers corresponding to the sequence of productions used in a leftmost derivation of  $w$ . A *right parse* of a sentence  $w \in L(G)$  is the string of numbers corresponding to the sequence of productions used in a rightmost derivation of  $w$ , but given in the reversed order.  $N \cup T$  is denoted by  $V$ . The empty string is denoted by  $\epsilon$ . If  $\alpha \in V^*$ , then the *length* of  $\alpha$  is denoted by  $l(\alpha)$  and the *reverse* of  $\alpha$  is denoted by  $\alpha^R$ . The notation  $A \Rightarrow^\pi \alpha$  is used to show that the derivation of  $\alpha$  from  $A$  can be done according to the sequence of productions  $\pi$ . Hence, if  $S \Rightarrow_L^\pi w$  (a leftmost derivation), then  $\pi$  is a left parse for  $w$  and if  $S \Rightarrow_R^\pi w$  (a rightmost derivation), then  $\pi^R$  is a right parse for  $w$ .

We assume that the grammars in this paper are *reduced*, that is, for any  $X \in V$  we have  $S \Rightarrow^* \alpha X \beta \Rightarrow^* w$  for some  $\alpha, \beta \in V^*$  and  $w \in T^*$ .

**Definition 1.** A context-free grammar  $G = (N, T, P, S)$  is said to be

- (i) *left-regular* if  $P \subset N \times (N \cup \{\epsilon\}) T$ ,

- (ii) *right-regular* if  $P \subset N \times T(N \cup \{\epsilon\})$ ,
- (iii) in *Greibach normal form* if  $P \subset (N \times TN^*) \cup \{(S, \epsilon)\}$ .

Now we consider the cover-relation between grammars. In the following definition of cover left parses are mapped on right parses.

**Definition 2.** A context-free grammar  $G' = (N', T, P', S')$  is said to *left-to-right cover* the context-free grammar  $G = (N, T, P, S)$  if there exists a homomorphism  $\varphi : P' \rightarrow P^*$  such that

- (i) if  $\pi'$  is a left parse for a sentence  $w$  with respect to  $G'$ , then  $\varphi(\pi')$  is a right parse for  $w$  with respect to  $G$ ,
- (ii) for each right parse  $\pi$  of a sentence  $w$  with respect to  $G$ , there exists a left parse  $\pi'$  for  $w$  with respect to  $G'$  such that  $\varphi(\pi') = \pi$ .

Variants of this definition can be found in [1,3,7].

Analogous definitions can be given for right covers, in which case right parses are mapped on right parses, for left covers (left parses on left parses) and for right-to-left covers (right parses on left parses). From the definition it follows that  $L(G') = L(G)$ .

The cover-homomorphism  $\varphi$  is said to be *fine* if  $\varphi : P' \rightarrow P \cup \{\epsilon\}$  (in [4] this is called a production map), and *very fine* if  $\varphi : P' \rightarrow P$ .

Notice that with our definition productions are mapped on (possibly empty) strings of productions. The right cover definition of Gray and Harrison [3] can be compared with ours if we demand that in our definition  $\varphi$  is a fine homomorphism, hence productions are mapped on productions or on the empty string.

In the definition of Aho and Ullman [1]  $\varphi$  should also be considered as a fine cover-homomorphism. Since our definition is slightly more general some cover results can be obtained which are not obtainable with the definitions which make use of a fine cover-homomorphism.

**Example 1.** Let  $G$  be the grammar with only productions  $S \rightarrow AB$ ,  $A \rightarrow a$  and  $B \rightarrow b$ . Any context-free grammar  $G'$  in Greibach normal form for  $L(G) = \{ab\}$  has at least two productions. Sentence  $ab$  of grammar  $G'$  in Greibach normal form is generated with exactly two productions, one of the form  $S' \rightarrow aC$  and one of

the form  $C \rightarrow b$ . Hence, a cover-homomorphism has to map a string of two productions on a string of three productions which means that the homomorphism can not be fine. However, any cover (left, right, left-to-right, right-to-left) can be defined if we use an 'arbitrary' cover-homomorphism.

Obviously, the observation in Example 1 remains valid if we take the definition of Greibach normal form such that  $P \subset (N \times TV^*) \cup \{(S, \epsilon)\}$  as is done in [3].

Theorem 1.4 in [3] states that the grammar  $G_0$  with only productions  $S \rightarrow S_0$ ,  $S \rightarrow S_1$ ,  $S \rightarrow 0$  and  $S \rightarrow 1$  can not be right covered (under a fine cover-homomorphism) by a grammar in Greibach normal form. It is the aim of this paper to show that with our definition each left-regular grammar (hence, also  $G_0$ ) can be right covered and left-to-right covered by a Greibach normal form grammar.

A few notes on notations are in order. We use the notation  $i.A \rightarrow \alpha$  for a production  $A \rightarrow \alpha$  with label  $i$ . In a situation where production  $A \rightarrow \alpha$  is mapped on a string  $\pi$  of productions we sometimes use  $A \rightarrow \alpha(\pi)$ . Moreover,

- (i)  $G' [l/\bar{r}] G$ , if  $G'$  left-to-right covers  $G$ ,
- (ii)  $G' [\bar{r}/\bar{r}] G$ , if  $G'$  right covers  $G$ ,
- (iii)  $G' [\bar{r}/l] G$ , if  $G'$  right-to-left covers  $G$ , and
- (iv)  $G' [l/l] G$ , if  $G'$  left covers  $G$ .

## 2. Covers for left-regular grammars

The two transformations presented in this section are very simple, therefore we omit detailed proofs. The first algorithm is a well-known method to obtain a right-regular grammar from a left-regular grammar.

### Algorithm 1.

*Input.* A left-regular grammar  $G = (N, T, P, S)$ .

*Output.* A right-regular grammar  $G' = (N', T, P', S')$  which left-to-right covers  $G$  under a very fine cover-homomorphism.

*Method.* Each production in  $P'$  is followed by its image under the cover-homomorphism. Initially set  $P' = \emptyset$ .

Which elements are in  $N'$  will be obvious.

- (i) Let  $A \in N$ ,  $A \neq S$ , then, if  $i.A \rightarrow a$  in  $P$ , then add  $S' \rightarrow aA$  (i) to  $P'$ , and if  $i.A \rightarrow Ba$  in  $P$ , then add

$B \rightarrow aA$  (i) to  $P'$ .

(ii) If  $i.S \rightarrow a$  is in  $P$ , add  $S' \rightarrow a$  (i) to  $P'$ . Moreover, if  $S$  is left recursive, add  $S' \rightarrow aS$  (i) to  $P'$ . If  $i.S \rightarrow Aa$  is in  $P$ , add  $A \rightarrow a$  (i) to  $P'$ . Moreover, if  $S$  is left recursive, add  $A \rightarrow aS$  (i) to  $P'$ .

**Lemma 1.**  $G' [l/\bar{r}] G$  under a very fine cover-homomorphism  $\psi$ .

**Proof.** In the proof we use two claims. Homomorphism  $\psi$  which is used has already been defined in Algorithm 1. Clearly,  $\psi$  is very fine.

**Claim 1.** Assume  $B \neq S'$ . If  $B \Rightarrow_L^{\pi'} wA$  in  $G'$ , then there exists a derivation  $A \Rightarrow_R^{\pi} Bw$  in  $G$  with  $\psi(\pi') = \pi^R$ .

**Proof of Claim 1.** Induction on  $l(w)$ . Assume  $l(w) = 1$ , hence  $w \in T$  and we write  $w = a$  and  $\pi' = i'$ . Since  $i'$  is  $B \rightarrow aA$  in  $P'$  there exists  $i.A \rightarrow Ba$  in  $P$  with  $\psi(i') = i$ . Now assume  $l(w) > 1$ . If we write  $w = au$  and  $\pi' = i'p'$  with  $i'.B \rightarrow aC$  in  $P'$ , then we have a derivation

$$B \xRightarrow[L]{i'} aC \xRightarrow[L]{\rho'} auA$$

in  $G'$ . Since  $C \Rightarrow_L^{\rho'} uA$  we may conclude from the induction hypothesis that  $A \Rightarrow_R^{\rho} Cu$  in  $G$ , where  $\psi(\rho') = \rho^R$ . Moreover,  $\psi(i') = i$  for the production  $i.C \rightarrow Ba$  in  $P$  and it follows that  $A \Rightarrow_R^{\pi} Bw$  in  $G$ , with  $\pi = \rho i$  and  $\psi(\pi') = \pi^R$ . This concludes the proof of Claim 1.

**Claim 2.** If  $A \neq S$  in the case that  $S$  is not left-recursive in  $G$ , then  $A \Rightarrow_R^{\pi} Bw$  in  $G$  implies that there exist  $\pi'$  such that  $B \Rightarrow_L^{\pi'} wA$  in  $G'$  and  $\psi(\pi') = \pi^R$ .

**Proof of Claim 2.** Observe that if  $S$  is left recursive in  $G$ , then the existence of a derivation  $S \Rightarrow_R^{\pi} Bw$  in  $G$  implies the existence of derivations  $B \Rightarrow_L^{\pi'} wS$  and  $B \Rightarrow_L^{\pi''} w$  in  $G'$  with  $\psi(\pi') = \psi(\pi'') = \pi^R$ . However, if  $S$  is not left-recursive, then there exists only the derivation  $B \Rightarrow_L^{\pi''} w$ . Therefore we have excluded this case from the claim.

The formal proof of Claim 2 proceeds again by induction on  $l(w)$  and since it goes along the same lines as the proof of Claim 1 we omit it.

Now consider a derivation  $S' \Rightarrow_L^{\pi'} w$  in  $G'$ . This

derivation can be written as

$$S' \xRightarrow[L]{i'} aA \xRightarrow[L]{\rho'} auB \xRightarrow[L]{j'} aub$$

where  $w = aub$  and  $\pi' = i'\rho'j'$ . From the construction of  $G'$  it follows that there exist  $j.S \rightarrow Bb$  and  $i.A \rightarrow a$  in  $P$ , with  $\psi(i') = i$  and  $\psi(j') = j$ . From Claim 1 it follows that  $B \Rightarrow_R^{\rho} Au$ , with  $\psi(\rho') = \rho^R$ . Hence,  $S \Rightarrow_R^{\pi} w$ , with  $\pi = j\rho i$  and  $\psi(\pi') = \pi^R$ .

Now consider the second condition of the cover definition. Assume  $S \Rightarrow_R^{\pi} w$  in  $G$ . We can write

$$S \xRightarrow[R]{j} Ab \xRightarrow[R]{\rho} Bub \xRightarrow[R]{i} aub,$$

where  $w = aub$  and  $j\rho i = \pi$ . Notice that if  $S$  is not left-recursive, then  $A \neq S$ . Hence, from Claim 2 it follows that  $B \Rightarrow_L^{\rho'} uA$  in  $G'$  with  $\psi(\rho') = \rho^R$ . From the construction of  $G'$  it follows that there exist  $i'.S' \rightarrow aB$  and  $j'.A \rightarrow b$  in  $P'$ , with  $\psi(i') = i$  and  $\psi(j') = j$ . Hence,  $S' \Rightarrow_L^{\pi'} w$ , with  $\pi' = i'\rho'j'$  and  $\psi(\pi') = \pi^R$ . This concludes the proof of Lemma 1.

Now we show that it is also possible to obtain a context-free grammar in Greibach normal form which also right covers the original left-regular grammar. This is done with a simple transformation on the right-regular grammar which is obtained in Algorithm 1. That is, if  $G' [l/\bar{r}] G$ , then  $G'$  will be transformed to a grammar  $G''$  in Greibach normal form such that  $G'' [\bar{r}/l] G'$ . Then we may conclude  $G'' [\bar{r}/\bar{r}] G$ .

The idea of the transformation is illustrated in Fig. 1. Notice that with this transformation the cover-homomorphism  $\psi$  is *not fine* anymore.

**Algorithm 2.**

**Input.** A right-regular grammar  $G_0 = (N_0, T, P_0, S)$ .

**Output.** A context-free grammar  $G_1 = (N_1, T, P_1, S)$  in Greibach normal form which right-to-left covers  $G_0$  under a cover-homomorphism  $\psi : P_1 \rightarrow P_0$ .

**Method.** Initially  $P_1$  is empty and  $N_1 = N_0$ . The

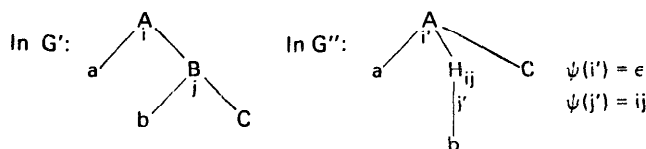


Fig. 1. Transformation from  $G'$  to  $G''$ ,  $\psi$  is not fine.

symbols of the form  $H_{ij}$  which are introduced during the algorithm are new nonterminal symbols which are added to  $N_1$ .

In this algorithm we let the productions in  $P_1$  follow by their images under the cover-homomorphism  $\psi$ . The productions in  $P_0$  are labeled from 1 to  $|P_0|$ :

(i) For each production of the form  $i.A \rightarrow aB$  in  $P_0$  the following is done. Let  $j_p.B \rightarrow b_p C_p$  for  $1 \leq p \leq m$  and  $k_q.B \rightarrow c_q$  for  $1 \leq q \leq n$  be all productions in  $P_0$  with left-hand side  $B$ . Then add to  $P_1$  the following productions:

$$A \rightarrow aH_{ij_1}C_1 \mid \dots \mid aH_{ij_m}C_m(\epsilon, \dots, \epsilon) \quad \text{and}$$

$$H_{ij_p} \rightarrow b_p(ij_p) \quad \text{for } 1 \leq p \leq m;$$

$$A \rightarrow aH_{ik_1} \mid \dots \mid aH_{ik_n}(\epsilon, \dots, \epsilon) \quad \text{and}$$

$$H_{ik_q} \rightarrow c_q(ik_q) \quad \text{for } 1 \leq q \leq n.$$

(ii) For each production in  $P_0$  of the form  $l.A \rightarrow a$  add to  $P_1$  the production  $A \rightarrow a(l)$ .

(iii) Remove all useless productions and nonterminals of the grammar which is obtained in step (i) and (ii).

*End of the algorithm.*

Since the transformation is simple the following observations on the algorithm are adequate to decide that  $G_1[\bar{r}/l] G_0$  (cf. also the simple example in Fig.2). Consider a derivation  $S \Rightarrow_L^r w$  in  $G_1$ . Obviously, there exists  $S \Rightarrow_R^r w$  in  $G_1$ , such that  $\pi$  and  $\rho$  describe the same parse tree. Define a homomorphism  $\phi : P_1 \rightarrow P_1$  such that  $\phi(p) = p$  if  $p$  is of the form  $Q \rightarrow a$  with  $Q \in N_1$  and  $a \in T$ , and  $\phi(p) = \epsilon$  otherwise. It is not difficult to verify that for the derivations mentioned above  $\phi(\pi) = \phi(\rho^R)$ . Since  $\psi(p) = \epsilon$  if  $\phi(p) = \epsilon$  it follows that, with our definition of  $\psi$ ,  $G_1[\bar{r}/l] G_0$  if and only if  $G_1[l/l] G_0$ .

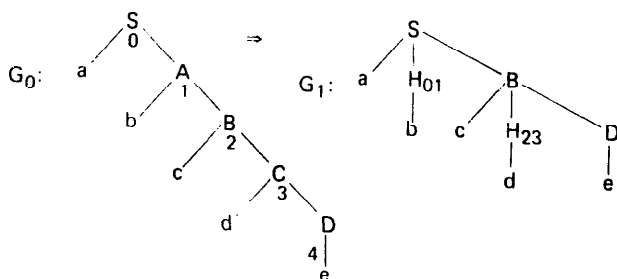


Fig. 2. Tree structures in  $G_0$  and  $G_1$ .

If  $S \Rightarrow_L^r w$  in  $G_1$  then  $\phi(\pi')$  is of the form  $m_0 m_1 \dots m_n p$  with  $m_k, H_{i_k j_k} \rightarrow b_k$  and  $p = \epsilon$  or  $p.A \rightarrow b$  with  $A \in N_0, b$  and  $b_k \in T$  and  $H_{i_k j_k} \in N_1 - N_0$ . We have

$$\psi(\pi') = \psi(\phi(\pi')) = i_0 j_0 i_1 j_1 \dots i_n j_n p$$

and this string is a leftmost derivation for  $w$  in  $G_0$ . If  $S \Rightarrow_L^r w$  in  $G_0$ , then likewise we can find a derivation  $S' \Rightarrow_L^r w$  in  $G_1$  with  $\psi(\pi') = \pi$ . Therefore  $G_1[l/l] G_0$  and also  $G_1[\bar{r}/l] G_0$ .

**Corollary 1.** Algorithm 2 transforms each right regular grammar  $G_0$  to a grammar  $G_1$  in Greibach normal form such that  $G_1[l/l] G_0$  and  $G_1[\bar{r}/l] G_0$ .

From the transitivity of the cover-relation we obtain the following result.

**Corollary 2.** Each left-regular grammar is right covered by a context-free grammar in Greibach normal form.

**Notes.** Instead of Algorithm 1 it is also possible to use the Rosenkrantz-method [11]. This method, applied to a left-regular grammar yields the same cover result.

The methods which are used here can be generalized and used in other situations to obtain cover results for more general classes of grammars. These results will appear in a forthcoming paper (cf. [9]).

**Acknowledgment**

I am grateful to the referees for some corrections and their suggestions regarding the presentation of these results.

**References**

- [1] A.V. Aho and J.D. Ullman, *The Theory of Parsing, Translation and Compiling*, Vols. 1 and 2 (Prentice-Hall, Englewood Cliffs, NJ, 1972 and 1973).
- [2] J. Gray and M.A. Harrison, *Single pass precedence analysis*, IEEE Conf. Record of the 10th Ann. Symp. on Switching and Automata Theory (1969) 106-117.
- [3] J. Gray and M.A. Harrison, *On the covering and reduction problems for context-free grammars*, J. ACM 19 (1972) 675-698.

- [4] H.B. Hunt, D.J. Rosenkrantz and T.G. Szymanski, The covering problem for linear context-free grammars, *Theoret. Comput. Sci.* 2 (1976) 361–382.
- [5] H.B. Hunt, D.J. Rosenkrantz and T.G. Szymanski, On the equivalence, containment, and covering problems for the regular and context-free languages, *J. Comput. System Sci.* 12 (1976) 222–268.
- [6] A. Nijholt, Cover results and normal forms, in: J. Gruska, Ed., *Mathematical Foundations of Computer Science 1977*, Lecture Notes in Computer Science 53 (Springer, Berlin, 1977) 420–429.
- [7] A. Nijholt, On the covering of parsable grammars. *J. Comput. System Sci.* 15 (1977) 99–110.
- [8] A. Nijholt, On the covering of left-recursive grammars, *Proc. 4th ACM-SIGACT/SIGPLAN Symp. on Principles of Programming Languages* (1977) 86–96.
- [9] A. Nijholt, A survey of normal form covers for context-free grammars, IR-49, Vrije Universiteit, Amsterdam (1979).
- [10] J.C. Reynolds, Grammatical covering, Argonne National Laboratory T.M. No. 96 (1968).
- [11] J.C. Reynolds and R. Haskell, Grammatical coverings, unpublished manuscript (1970).
- [12] D.J. Rosenkrantz, Matrix equations and normal forms for context-free grammars, *J. ACM* 14 (1967) 501–607.
- [13] E. Soisalon-Soininen, On the covering problem for left-recursive grammars, *Theoret. Comput. Sci.* 8 (1979) 1–12.
- [14] E. Ukkonen, The nonexistence of some covering context-free grammars, *Information Processing Lett.* 8 (4) (1978) 187–192.