WEIGHTREDUCING GRAMMARS AND ULTRALINEAR LANGUAGES

ULRIKE BRANDT\textsuperscript{1}, GHISSLAIN DELEPINE\textsuperscript{1} AND HERMANN K.-G. WALTER\textsuperscript{1}

Abstract. We exhibit a new class of grammars with the help of weightfunctions. They are characterized by decreasing the weight during the derivation process. A decision algorithm for the emptiness problem is developed. This class contains non-contextfree grammars. The corresponding language class is identical to the class of ultralinear languages.

Mathematics Subject Classification. 68Q45.

INTRODUCTION

The emptiness problem for classes of grammars containing non-contextfree grammars is in general difficult to solve. The reader should remember that this problem is undecidable for context-sensitive grammars. Moreover the word problem can be reduced to the emptiness problem under very mild conditions. We exhibit a class of grammars with a solvable emptiness problem, which contains non-contextfree grammars. Our method uses weightfunctions such that the weight decreases during the derivation process, moreover a criterion is added, which separates via the weightfunction variables and terminals. This class of grammars is called the class of weightreducing grammars. For this class we develop a decision algorithm for the emptiness problem. Furthermore we show that the corresponding language family is exactly the family of ultralinear languages.

Keywords and phrases. Chomsky-grammars, weightfunctions, weightreducing grammars, emptiness problem, ultralinear languages.

\textsuperscript{1} University of Technology, Darmstadt Department of Computer Science, Germany; e-mail: [brandt; walter]@iti.informatik.tu-darmstadt.de, gdelepine@yahoo.fr

© EDP Sciences 2004
1. Basic notations and definitions

Let $X$ be an alphabet, then $X^*$ is the set of words $w$ over $X$ (free monoid). $\Box$ is the empty word and $X^* = X^* \setminus \Box$. Fixing $x \in X$ we define the homomorphism $|w|_x : X^* \to \mathbb{N}$ by $|y|_x = \delta_{x,y}$ ($y \in X, \delta_{x,y}$ is Kronecker’s symbol), hence $|w|_x$ is the number of occurrences of $x$ in $w$.

For $X' \subseteq X$ we define: $|w|_{X'} = \sum_{x' \in X'} |w|_{x'}$, therefore $|w|_X = |w|$ is the length of $w$.

A (Chomsky-)grammar $G$ is a quadruple $G = (V,T,P,\sigma)$ where $V,T$ are alphabets with $V \cap T = \emptyset$, $\sigma \in V$ and $P \subseteq V^+ \times (V \cup T)^+ \times V$ is a finite set.

We call $V$ the set of variables, $T$ the set of terminals, $A = V \cup T$ the alphabet of $G$, $\sigma$ the startsymbol and $P$ the set of productions. As usual $(p,q) \in P$ will be written $p \rightarrow q$.

With respect to the underlying Semi-Thue-System $(A,P)$ we define derivations of words in the following way. For every $w,w' \in A^*$ we write $w \vdash w'$ iff there exist $u,v \in A^*$, $p \rightarrow q \in P$ such that $w = upv$ and $w' = uqv$. $w \vdash w'$ is the reflexive and transitive closure of $\vdash$.

For every grammar $G$ the generated language $L(G)$ is defined by

$$L(G) = \{ w \in T^* \mid \sigma \vdash^* w \}.$$ 

Grammar classes are denoted by $\Gamma$ and the associated language family is $L(\Gamma) = \{ L \mid \exists G \in \Gamma : L(G) = L \}$.

We are mostly interested in the following grammar classes:

- $\Gamma_{Ch} = \text{all Chomsky-grammars};$
- $\Gamma_{cf} = \{ G \in \Gamma_{Ch} \mid \forall p \rightarrow q \in P : |p| = 1 \};$
- $\Gamma_{lin} = \{ G \in \Gamma_{cf} \mid \forall p \rightarrow q \in P : q \in T^* \cdot (V \cup \Box) \cdot T^* \};$
- $\Gamma_{lin.index} = \{ G \in \Gamma_{Ch} \mid \exists k \in \mathbb{N} \forall w \in L(G) \exists n = u_0 \vdash u_1 \vdash \ldots \vdash u_n = w \forall 0 \leq i \leq n : |u_i|_V \leq k \} \text{ (see [1]);}$
- $\Gamma_{ultrilinear} = \{ G \in \Gamma_{cf} \mid \exists a \text{ partition } (A_i)_{i=1}^n \text{ of } V, \forall i \in [1 \ldots n], \xi \in A_i : \xi \rightarrow p \in P \Rightarrow p \in (T \cup \bigcup_{k=0}^{i-1} A_k)^* \cup T^* \cdot A_i \cdot T^* \} \text{ (see [4]).}$

The corresponding language families are $\mathcal{L}_{Ch}, \mathcal{L}_{cf}, \mathcal{L}_{lin}, \mathcal{L}_{lin.index}$ and $\mathcal{L}_{ultrilinear}$.

We assume the reader to be familiar with the basic concepts of grammars and languages (see [5,6]).

2. Weightreducing grammars

**Definition 2.1.** Let $G \in \Gamma_{Ch}, \gamma : A^* \to \mathbb{N}$ a homomorphism.

$\gamma$ reduces $G$ iff

(i) $\forall p \rightarrow q \in P : \gamma(p) \geq \gamma(q)$;

(ii) $\forall x \in A : \gamma(x) = 0 \Leftrightarrow x \in T$.

**Definition 2.2.** A grammar $G$ is weightreducing iff there is a homomorphism $\gamma$ that reduces $G$. 

The class of weightreducing grammars is denoted by $\Gamma_{\text{wr}}$ and $L_{\text{wr}}$ is the associated language family.

**Remark.** Our definition is something of a counterpart of context-sensitive grammars. For context-sensitive grammars the weight is increasing.

**Observation 2.1.**

(i) $w \Rightarrow^* w' \Rightarrow \gamma(w) \geq \gamma(w')$;

(ii) $\sigma \Rightarrow^* w \Rightarrow |w|_V \leq \gamma(\sigma)$.

**Example 2.1.** For any $G \in \Gamma_{\text{lin}}$ let $\gamma(\xi) = 1$ for all $\xi \in V$. Then $\gamma$ reduces $G$.

**Example 2.2.** Consider for any $k \geq 1$ the grammar $G_{1,k}$ with $\sigma = \sigma_k$ and the set of productions

$\sigma_{k-i} \rightarrow (\sigma_{k-i}) | (\sigma_{k-i-1}) \sigma_{k-i-1} | \Box$ ($0 \leq i \leq k - 2$)

$\sigma_1 \rightarrow (\sigma_1) | \Box$.

Choose: $\gamma(\sigma_{k-i}) = 2^{k-i}(0 \leq i < k)$ then $\gamma$ reduces $G$.

Observe that with the help of $D_{1,k} = L(G_{1,k})$ the index-hierarchy is shown in [4].

**Example 2.3.** Consider the grammar $G$ with $\sigma \rightarrow \sigma c \xi | \Box, \xi \rightarrow a \xi b | \Box$, then $L(G) = (c \cdot \{a^n b^n \mid n \geq 1\})^*$. $G$ is a finite-index grammar, but not weightreducing.

Since $L_{\text{fin}} \subseteq L_{\text{wr}}$ by Example 2.1 and $L_{\text{fin, index}} \subseteq L_{\text{cf}}$ by the Ginsburg-Spanier-theorem [3] we conclude $L_{\text{fin, index}} \subseteq L_{\text{wr}} \subseteq L_{\text{fin}} \subseteq L_{\text{cf}}$ by Observation 2.1(ii).

We now study the question, how reducing $\gamma$’s can be calculated.

**Theorem 2.1.** The question whether a grammar allows a reducing function, i.e. is a weightreducing grammar or not, is decidable.

**Proof.** Let $G$ be a grammar with $V = \{\xi_1, \ldots, \xi_n\}$ and $\sigma = \xi_1$. Since by condition (ii) of Definition 2.1 a possible $\gamma$ must automatically fulfil $\gamma(x) = 0$ for $x \in T$, only the $\gamma(\xi_i)$ have to be determined. But then conditions (i) and (ii) of Definition 2.1 rewrite to

(1) $p \rightarrow q \in P \Rightarrow \sum_{i=1}^n (|p|_{\xi_i} - |q|_{\xi_i}) \cdot \gamma(\xi_i) \geq 0$;

(2) $\gamma(\xi_i) > 0$ for $1 \leq i \leq n$.

Therefore the construction of a reducing $\gamma$ is equivalent to solve the following system of linear inequations with variables $x_1, \ldots, x_n$ over $\mathbb{Q}$:

$$\sum_{i=1}^n (|p|_{\xi_i} - |q|_{\xi_i}) \cdot x_i \geq 0 \quad (p \rightarrow q \in P) \quad \text{and} \quad x_i > 0 (1 \leq i \leq n).$$

If $\gamma$ is reducing then $x_i = \gamma(\xi_i) (1 \leq i \leq n)$ is a solution, conversely if $(x_1, \ldots, x_n)$ is a solution then defining $\gamma(\xi_i) = \lambda x_i$ for $1 \leq i \leq n$ and suitable $\lambda \in \mathbb{N}$ we obtain a reducing $\gamma$.\qed
3. Ultralinear and Weightreducing Grammars

We want to show: \( \mathcal{L}_{\text{ultralinear}} = \mathcal{L}_{\text{wr}} \). To do this we study certain transformations of grammars. The following definitions introduced in [2] are useful:

**Definition 3.1.** For every \( G \in \Gamma_{\text{Ch}} \) and for every \( w \in A^* \) the rank of \( w \) \( r(w) \) is defined by \( r(w) = \sup \{|u|_V \mid u \in A^* \text{ and } w \vdash^* u \} \).

**Observation 3.1.**
(i) If \( G \in \Gamma_{\text{Ch}} \) then: \( w_1, w_2 \in A^* \Rightarrow r(w_1 w_2) \geq r(w_1) \cdot r(w_2) \).
(ii) If \( G \in \Gamma_{\text{cf}} \) then: \( w_1, w_2 \in A^* \Rightarrow r(w_1 w_2) = r(w_1) \cdot r(w_2) \).

**Definition 3.2.** A grammar \( G \in \Gamma_{\text{Ch}} \) is **variable-bounded** iff there exists a constant \( k \in \mathbb{N} \) such that for every \( w \in A^* : \sigma \vdash^* w \Rightarrow |w|_V \leq k \).

**Theorem 3.1.** If \( G \in \Gamma_{\text{Ch}} \) is weightreducing then \( G \) is variable-bounded.

**Proof.** Let \( G \in \Gamma_{\text{cf}} \) be weightreducing and \( \gamma \) the corresponding weightfunction. Suppose \( G \) is not variable-bounded. Consider \( k = \gamma(\sigma) \) and a word \( w \in A^* \) with \( \sigma \vdash^* w \) and \( |w|_V > k \). But then \( \gamma(w) \geq |w|_V > k \geq \gamma(\sigma) \), a contradiction to Observation 2.1(ii). □

A variable \( \xi \in V \) is **reachable** from \( \sigma \) iff \( \sigma \vdash^* u\xi v \) for some \( u, v \in A^* \).

**Theorem 3.2.** If \( G \in \Gamma_{\text{cf}} \) is variable-bounded and every variable is reachable from \( \sigma \) then \( G \) is weightreducing.

**Proof.** Let \( G \in \Gamma_{\text{cf}} \) be variable-bounded by \( k \) and every \( \xi \in V \) reachable from \( \sigma \). In this case the rank \( r \) has the property \( r(\xi) \leq k \) for every \( \xi \in V \). Furthermore by definition of \( r \), \( r(x) = 0 \) for every \( x \in T \). Hence, \( r \) is a reducing function for \( G \) because Observation 3.1(ii) ensures that \( r \) is a homomorphism in the context-free case. □

Combining Theorems 3.1 and 3.2 we get

**Theorem 3.3.** If \( G \in \Gamma_{\text{cf}} \) and every \( \xi \in V \) is reachable from \( \sigma \) then \( G \) is variable-bounded iff \( G \) is weightreducing.

**Theorem 3.4.** The family of ultralinear languages coincides with the family of contextfree weightreducing languages.

**Proof.** In [2] is shown: If \( G \in \Gamma_{\text{cf}} \) then \( G \) is ultralinear iff \( G \) is variable-bounded. □

Theorem 3.4 doesn’t transfer directly to \( \mathcal{L}_{\text{wr}} \). This is due to the fact that the rank of \( G \in \Gamma_{\text{Ch}} \) is in general not a homomorphism and Theorem 3.2 does not hold in the general case if \( G \) is any Chomsky-grammar.
Consider for example the grammar $G$ given by

$$
\begin{align*}
\sigma & \rightarrow \xi\beta \\
\xi\beta & \rightarrow a\xi b\beta c\gamma d \\
\xi & \rightarrow a \\
\beta & \rightarrow b \\
\gamma & \rightarrow c.
\end{align*}
$$

$G$ is variable-bounded with $k = 3$ but not weightreducing.

But there is another way to show $L_{\text{ultralinear}} = L_{\text{wr}}$ and that we prove $L_{\text{wr}} = L(\Gamma_{\text{cf}} \cap \Gamma_{\text{wr}})$ using a construction similar to the one showing $L_{\text{fin}, \text{index}} = L(\Gamma_{\text{cf}} \cap \Gamma_{\text{fin}, \text{index}})$ found in [3].

For every alphabet $A$ and $k \in \mathbb{N}$ let $A \leq k = \{ w \in A^* \mid |w| \leq k \}$.

**Theorem 3.5.** The family of ultralinear languages coincides with the family of weightreducing languages.

**Proof.** Like mentioned above we show $L_{\text{wr}} = L(\Gamma_{\text{cf}} \cap \Gamma_{\text{wr}})$. Consider $G \in \Gamma_{\text{wr}}$.

Then $G$ is variable-bounded with $k = \gamma(\sigma)$ by Theorem 3.1.

Our aim is to replace every production $p \rightarrow q$ with $p \in V^+$ by a set of contextfree productions simulating $p \rightarrow q$. This is possible because there are only finitely many $x, y \in V^*$ such that $xpy$ occurs in a word derivable from $\sigma$. Every $xpy$ of this kind interpreted as a new single variable builds the left hand-side of a new production. Then we can show that the resulting contextfree grammar remains variable-bounded and generates the same language as $G$.

More precisely, given a word $w = v_0 x_1 v_1 \ldots x_n v_n$ with $n \geq 0$, $v_i \in V^*$ ($0 \leq i \leq n$), $x_i \in T$ ($1 \leq i \leq n$), associate to it a new word $f(w)$ defined by $f(w) = \langle v_0 \rangle x_1 \langle v_1 \rangle \ldots x_n \langle v_n \rangle$. Identify $\langle \Box \rangle$ with the empty word $\Box$. Then $f(w)$ is defined over the new alphabet $T \cup \langle V \leq k \rangle$. Note that if a set $M$ of words over $A$ is “variable-bounded” in the sense that $|w|_V \leq k$ for every $w \in M$, the new set of words $f(M)$ is defined over $T \cup \langle V \leq k \rangle$ and this alphabet is finite.

Now, define the new contextfree grammar $G'$ by

$$
T' = T, \quad V' = \langle V \leq k \rangle \setminus \langle \Box \rangle, \quad \sigma' = \langle \sigma \rangle
$$

and

$$
P' = \{ (xpy) \rightarrow f(xqy) \mid p \rightarrow q \in P \text{ and } xy \in V \leq k - |p| \}.
$$

Clearly, $P'$ is finite, because $P$ is finite and $V \leq k - |p|$ is finite for every $p$ on the left hand-side of a production in $P$.

Furthermore, if $u \vdash w$ by some production in $G$, $f(u) \vdash f(w)$ by some production in $G'$ and vice versa.

Hence $\sigma \vdash^* w$ if and only if $f(\sigma) \vdash^* f(w)$ where $f(\sigma) = \langle \sigma \rangle = \sigma'$ showing $L(G) = L(G')$. 

It remains to show, that $G'$ is variable-bounded. Consider a derivation $\sigma' = \langle \sigma \rangle \vdash u$ in the new grammar $G'$. Then $u = f(w)$ for some $w \in A^*$, i.e. $\sigma \vdash w$ is a derivation in $G$. But then $|w|_V \leq k$, since $G$ is variable-bounded with $k$. By construction $|u|_V = |f(w)|_V \leq |w|_V \leq k$, i.e. $G'$ is variable-bounded with the same $k$ as $G$ and the statement follows directly by Theorem 3.2. □

**Corollary.** The emptiness problem for $\Gamma_{wr}$, i.e. the question whether a grammar $G \in \Gamma_{wr}$ generates the empty set or not, is decidable.

**Proof.** Let $G \in \Gamma_{wr}$ and $\gamma$ the weightfunction. If $\gamma$ is not given compute it by Theorem 2.1. Then the following algorithm decides if $L(G) = \emptyset$: Let $k = \gamma(\sigma)$.

1. Construct the corresponding contextfree and weightreducing grammar $G'$ by Theorem 3.5 with $|P'| \leq |V|^{\leq k} \cdot |P|$.
2. Decide if $\sigma' \vdash w$ for some $w \in T^*$. This may be done with the help of the following algorithm:
   2.1 construct the grammar $G''$ from $G'$ replacing every terminal in every production by the empty word;
   2.2 construct the directed graph with nodes from $\langle V^{\leq k} \rangle^{\leq k}$ such that two nodes $u$ and $v$ are connected by an edge if and only if $v$ is directly derivable from $u$ by a production of $G''$;
   2.3 decide if there is a path from $\langle \sigma \rangle$ to the empty word. □

4. **Closing remarks**

We haven’t discussed any complexity question for the possible algorithms. The suggested approach to the emptiness-problem for weightreducing grammars involves:

(i) the solution of a (special) system of linear inequalities over $\mathbb{Q}$;
(ii) the construction of a specific directed graph associated to the grammar under inspection;
(iii) solving a specific pathproblem for this graph.

The last problem depends heavily on the size of the constructed graph, so this would be the crucial point.

**References**


Communicated by J. Berstel.
Received May 15, 2003. Accepted November 7, 2003.

To access this journal online:
www.edpsciences.org