# Nonterminal Separating Macro Grammars

Jan Anne Hogendorp*

*Department of Computer Science, University of Twente*
*P.O. Box 217, 7500 AE Enschede, The Netherlands*

We extend the concept of nonterminal separating (or NTS) context-free grammar to nonterminal separating *m*-macro grammar where the mode of derivation *m* is equal to ''unrestricted'', ''outside-in'' or ''inside-out''. Then we show some (partial) characterization results for these NTS *m*-macro grammars.

## 1. Introduction

Macro grammars have been introduced in [6,7] as a way to describe context-dependent aspects of the syntax of programming languages. They are an extension of context-free grammars generating, for each mode of derivation, a family of languages in between the families of context-free languages and of context-sensitive languages. Though outside-in (or *OI*-) macro languages are able to describe correctly the declaration and use of program variables, they have the disadvantage of possessing an **NP**-complete membership problem. For *IO*-macro languages the problem is roughly as complex as for context-free languages [1]; so it can be solved deterministically in polynomial time or in space $\log^2 n$. But *IO*-macro grammars seem to be less suitable for modeling the declaration of program variables.

Without considering this complexity issue any further we investigate in this paper a way to restrict macro grammars. It is inspired by a restriction on context-free grammars, viz. by the nonterminal separating (or NTS) condition [3]. For context-free grammars this restriction results in deterministic languages that have ''disjunct syntactic categories'' [3,5]. The actual NTS condition requires that adding the reductions corresponding to the productions of a grammar does not extend its set of sentential forms. Or, equivalently, the set of sentential forms does not change when we apply the rules of the grammar in both directions.

In Section 2 we provide the necessary notions, elementary results and terminology on macro grammars and on context-free grammars that satisfy the NTS condition. Section 3 is devoted to the definition of NTS macro grammar and some of their properties as far as they extend the corresponding results on NTS context-free grammars. We restrict our attention to characterization results of the NTS property for *m*-macro grammars where *m* is a mode of derivation, i.e., *m* equals either ''outside-in'' (or *OI*), ''inside-out'' (or *IO*) or ''unrestricted'' (or *UNR*). Finally, Section 4 contains some concluding remarks.

## 2. Preliminaries

### 2.1. Macro Grammars

Macro grammars have been introduced by Fischer in [6,7] as an extension of context-free grammars. In essence, they differ from context-free grammars in possessing a ranked alphabet of nonterminal symbols and so macro grammars are a particular kind of term rewriting system.

* The work of the author has been supported by the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

A *ranked alphabet* $\Delta$ is a finite set of symbols each of which is provided with a natural number, called its rank. For $i \geq 0$, let $\Delta_i$ denote the subalphabet of $\Delta$ that consists of all symbols of rank $i$. Thus if $i \neq j$, then $\Delta_i \cap \Delta_j = \emptyset$.

**Definition 2.1.1.** Let $\Delta$ be a ranked alphabet and PC the set of punctuation characters (i.e., left and right parenthesis and comma symbol). The *set* $T(\Delta)$ *of terms* over $\Delta$ is the smallest set of strings over $\Delta \cup \mathrm{PC}$ that satisfies

(i) $\Delta_0 \cup \{\lambda\} \subseteq T(\Delta)$; $\lambda$ denotes the empty word,

(ii) if $t_1, t_2 \in T(\Delta)$, then $t_1 t_2 \in T(\Delta)$,

(iii) if $A \in \Delta_n$ and $t_1, \ldots, t_n \in T(\Delta)$, then $A(t_1, \ldots, t_n) \in T(\Delta)$. □

Formally, we ought to write $A()$ if $A \in \Delta_0$; in practice we will omit the parentheses in that case. However, the notation $A(t_1, \ldots, t_n)$ does not imply that $n > 0$.

**Definition 2.1.2.** A *macro grammar* $G$ is a 5-tuple $G = (\Phi, \Sigma, X, P, S)$ where $\Phi$ is a ranked alphabet of *nonterminals*, $\Sigma$ is an alphabet of *terminals*, $X$ is a finite set of *variables* (Each terminal and variable has rank zero. The sets $\Phi$, $\Sigma$ and $X$ are disjoint.), $S \in \Phi_0$ is the *start symbol*, and $P$ is a finite set of *productions* or *rules* of the form $A(x_1, \ldots, x_n) \rightarrow t$ with $A \in \Phi_n$, $x_1, \ldots, x_n$ are mutually distinct elements of $X$, and $t$ is a term over $\Sigma \cup \Phi \cup \{x_1, \ldots, x_n\}$. □

Sentential forms of a macro grammar are terms over $\Sigma \cup \Phi$. Some specific subsets of terms give rise to interesting special types of macro grammars and corresponding sets of sentential forms. Viz. the set $BT(\Sigma \cup \Phi)$ of *basic terms* over $\Sigma \cup \Phi$ is the subset of $T(\Sigma \cup \Phi)$ of terms in which no $A \in \Phi$ appears in the argument list of another symbol of $\Phi$ (i.e., nonterminals are not nested). And the set $LBT(\Sigma \cup \Phi)$ of *linear basic terms* over $\Sigma \cup \Phi$ is the subset of $T(\Sigma \cup \Phi)$ of terms containing at most one nonterminal.

A production $A(x_1, \ldots, x_n) \rightarrow t$is called [*linear*] *basic* if $t$ is a [linear] basic term. A macro grammar is [*linear*] *basic* if all its productions are [linear] basic. A production $A(x_1, \ldots, x_n) \rightarrow t$ is called *argument preserving* if for each $i$ ($1 \leq i \leq n$), $t$ contains at least one occurrence of $x_i$, and it is called *non-duplicating* if $t$ contains at most one occurrence of $x_i$ for each $i$ ($1 \leq i \leq n$).

In order to describe several modes of derivation for macro grammars we need the following concepts.

**Definition 2.1.3.** Let $\sigma$ be a term over $\Sigma \cup \Phi$. $\tau$ is a *subterm* of $\sigma$ if $\tau$ is a term over $\Sigma \cup \Phi$ and $\tau$ is a substring of $\sigma$.

A subterm $\tau$ of $\sigma$ occurs at *top level* in $\sigma$ if there exist subterms $\sigma_1$ and $\sigma_2$ such that $\sigma = \sigma_1 \tau \sigma_2$. So $\tau$ does not appear within the argument list of some nonterminal in $\sigma$.

A term over $\Sigma \cup \Phi$ is called *expanded* if it contains no nonterminals together with its associated argument list, or equivalently, if it is a string over $\Sigma$. □

Using the productions of a macro grammar one can expand terms. As usual we distinguish three modes of derivation.

*Unrestricted mode* (*UNR*): An occurrence of a nonterminal together with its arguments can be expanded according to a production by replacing the nonterminal and its arguments by the right-hand side of that production in which the arguments have been substituted for the corresponding variables.

*Inside-Out* (*IO*): A nonterminal with its arguments is expanded only if its arguments are all expanded terms.

*Outside-In* (*OI*): A nonterminal with its arguments is expanded only if it occurs at top level.

Each of these modes of derivation gives rise to a derivation relation, formally defined as follows.

**Definition 2.1.4.** Let $G=(\Phi,\Sigma,X,P,S)$ be a macro grammer and let $\sigma,\tau\in T(\Sigma\cup\Phi)$. The relations $\Rightarrow_{UNR}$, $\Rightarrow_{IO}$ and $\Rightarrow_{OI}$ over $T(\Sigma\cup\Phi)$ are defined by
(1) $\sigma\Rightarrow_{UNR}\tau$ holds if $\sigma$ contains a subterm of the form $A(t_1,...,t_n)$ where $A\in\Phi_n$ and $t_1,...,t_n\in T(\Sigma\cup\Phi)$, $P$ contains a production $A(x_1,...,x_n)\rightarrow t$ and $\tau$ results from $\sigma$ by substituting $A(t_1,...,t_n)$ by $t[t_1/x_1,...,t_n/x_n]$.
(2) $\sigma\Rightarrow_{IO}\tau$ holds in case $\sigma\Rightarrow_{UNR}\tau$ and all the arguments of the rewritten nonterminal are expanded terms.
(3) $\sigma\Rightarrow_{OI}\tau$ holds in case $\sigma\Rightarrow_{UNR}\tau$ and the subterm of $\sigma$ which is rewritten occurs at top level in $\sigma$. □

Let $\Leftarrow_m$ be the converse of $\Rightarrow_m$, i.e., for all $\sigma,\tau\in T(\Sigma\cup\Phi)$, $\sigma\Leftarrow_m\tau$ holds if and only if $\tau\Rightarrow_m\sigma$. And let $\Leftrightarrow_m$ be the union of $\Rightarrow_m$ and $\Leftarrow_m$. The reflexive and transitive closures of $\Rightarrow_m$, $\Leftarrow_m$ and $\Leftrightarrow_m$ are denoted by $\Rightarrow^*_m$, $\Leftarrow^*_m$ and $\Leftrightarrow^*_m$, respectively. In case $\sigma\Leftarrow^*_m\tau$ [$\sigma\Leftarrow_m\tau$] we say that $\sigma$ *reduces* [*directly*] *to* $\tau$.

It is easy to see that $\Leftrightarrow^*_m$ is a congruence relation. Obviously, it is an equivalence relation and the congruency follows from: $\sigma\Leftrightarrow^*_m\tau$ and $\alpha\Leftrightarrow^*_m\beta$ imply $\sigma\alpha\Leftrightarrow^*_m\tau\beta$; for $m=UNR$ this is trivial and in the other cases it follows from the fact that concatenation does not cause any additional nesting.

**Definition 2.1.5.** Let $G$ be a macro grammar and $m$ a mode of derivation. An *m-macro grammar* is a pair $(G,m)$, or simply denoted by $G$ when $m$ is known from the context. The *language* generated by an $m$-macro grammar $G=(\Phi,\Sigma,X,P,S)$ is defined by

$$L_m(G)=\{w\in\Sigma^* | S\Rightarrow^*_m w\}.$$

By OI, IO and UNR we denote the family of languages generated by *OI*-, *IO*- and *UNR*-macro grammars, respectively. □

In [6] Fischer proved the equality OI = UNR, and the fact that IO and OI are incomparable.

In the sequel many of our results are restricted to macro grammars which possess the property that every term derived by the macro grammar has a derivation that ultimately yields a string over the terminal alphabet. These macro grammars are called admissible macro grammars [6]. This property is defined as follows.

**Definition 2.1.6.** A $m$-macro grammar $G=(\Phi,\Sigma,X,P,Z)$ with $Z\subseteq\Phi_0$ is *admissible* if either $\Phi=Z$ and $P=\emptyset$ or
(1) for each $A\in\Phi$, there exists a sentential form of $G$ in which $A$ occurs,
(2) for each $A\in\Phi_n$ $(n\geq 0)$ and each $\sigma_1,...,\sigma_n\in\Sigma^*$ there exists a string $w$ over $\Sigma$ such that $A(\sigma_1,...,\sigma_n)\Rightarrow^*_m w$. □

In [6] it is shown that for each $m$-macro grammar there exists an equivalent admissible $m$-macro grammar. For $m=IO$ every $(G,m)$ has an equivalent admissible subgrammar; for $m=OI$ the task to find such an admissible grammar is more elaborate.

**Example 2.1.7.** Let $L_0\subseteq\{0,1\}^*$ the language containing exactly those words in which the number of 1's is equal to $2^n$ for some $n\geq 0$. $L_0$ is generated by the *OI*-macro grammar $G=(\Phi,\Sigma,X,P,S)$ with $\Phi=\Phi_0\cup\Phi_1$, $\Phi_0=\{S,A\}$, $\Phi_1=\{B\}$, $X=\{x\}$, $\Sigma=\{0,1\}$ and $P$ consists of the rules

$$S\rightarrow B(A)$$
$$B(x)\rightarrow B(xx)\,|\,x$$
$$A\rightarrow 0A\,|\,A0\,|\,1$$

In [6] it has been shown that $L_0$ cannot be generated by any *IO*-macro grammar. □

## 2.2. The NTS Property for Context-Free Grammars

NTS or nonterminal separating grammars have been introduced by Boasson [3]. A context-free grammar possesses the NTS property if its set of sentential forms is invariant when we apply the rules in both directions, i.e., when we use apart from its productions the corresponding reductions too.

Let $G=(V,\Sigma,P,Z)$ be a context-free grammar with alphabet $V$, terminal alphabet $\Sigma$ ($\Sigma\subseteq V$), set of productions $P$, and start set $Z$ ($Z\subseteq V-\Sigma$). For each $\omega\in V^*$ we denote the set of words over $\Sigma$ derivable from $\omega$ by $G$ as

$$L(G,\omega)=\{w\in\Sigma^* \mid \omega\Rightarrow^* w\}.$$

We call this set the language generated by $G$ from $\omega$. The language generated by $G$ is

$$L(G)=\{w\in\Sigma^* \mid \exists S\in Z: S\Rightarrow^* w\}.$$

The set of sentential forms generated by $G$ from $\omega\in V^*$ is

$$\underline{L}(G,\omega)=\{\psi\in V^* \mid \omega\Rightarrow^* \psi\}.$$

The relations $\leq=$, $\leq=^*$, $\Longleftrightarrow$ and $\Longleftrightarrow^*$ are defined in a way similar to §2.1; however, historically they were first defined for context-free grammars [3].

The set of words over $V$ derivable from $\omega\in V^*$ by both productions and the corresponding reductions is

$$\underline{LR}(G,\omega)=\{\psi\in V^* \mid \omega\Longleftrightarrow^* \psi\}.$$

**Definition 2.2.1.** A context-free grammar $G=(V,\Sigma,P,Z)$ has the *NTS property* or is an *NTS grammar* if for all $A\in V-\Sigma$, $\underline{LR}(G,A)=\underline{L}(G,A)$. A language $L$ is called an *NTS language* if there exists an NTS grammar that generates $L$. □

**Proposition 2.2.2.** [3,5]. *Let $G=(V,\Sigma,P,Z)$ be an NTS grammar. Then for all $A$ and $B$ in $V-\Sigma$, either $\underline{L}(G,A)\cap\underline{L}(G,B)=\emptyset$ or $\underline{L}(G,A)=\underline{L}(G,B)$ holds.* □

This property motivates the name of the concept defined in 2.2.1. However, the converse of 2.2.2 does not hold; e.g. $\{a^n b^n \mid n\geq 1\}\cup\{a^n b^{2n} \mid n\geq 1\}$ is not an NTS language [5], but it is easy to show that this language can be generated by a grammar that possesses ‘‘disjunct syntactic categories’’.

On the other hand NTS grammars can be characterized in the following way.

**Theorem 2.2.3.** [5,10]. *Let $G=(V,\Sigma,P,Z)$ be a context-free grammar. $G$ has the NTS property if and only if for all $A,B\in V-\Sigma$ and for all $\alpha,\beta,u\in V^*$ the following implication holds:*

$$\textit{if } A\Rightarrow^*\alpha u\beta \textit{ and } B\Rightarrow^* u, \textit{ then } A\Rightarrow^*\alpha B\beta.$$ □

For further details of context-free NTS grammars and languages the reader is referred to [2,3,5,8,9,10].

# 3. The NTS Property for Macro Grammars

### 3.1. Definitions

We use the following notational conventions. Usually, $(\sigma_1,...,\sigma_n)$ is abbreviated to $(\vec{\sigma}_{(n)})$. The subscript $(n)$ is necessary to distinguish for example $A(\vec{x}_{(n)})$ and $B(\vec{x}_{(k)})$. Only if no confusion is possible we write $\vec{x}$. For $A \in \Phi$, $A(\vec{x})$ is the left-hand side of a production; so $A(\vec{x}) = A$ if $A \in \Phi_0$. In the sequel an $m$-macro grammar will have a finite set $Z$ $(Z \subseteq \Phi_0)$ of initial symbols of rank 0 instead of a single initial symbol; cf. the definition of NTS context-free grammar.

**Definition 3.1.1.** Let $G = (\Phi,\Sigma,X,P,Z)$ be an $m$-macro grammar. Then the *language* generated by $(G,m)$ is

$$L_m(G) = \{w \in \Sigma^* | \exists S \in Z : S \Rightarrow_m^* w\},$$

and for each $t \in T(\Sigma \cup X \cup \Phi)$,

$$L_m(G,t) = \{w \in (\Sigma \cup X)^* | t \Rightarrow_m^* w\},$$
$$\underline{L}_m(G,t) = \{\omega \in T(\Sigma \cup X \cup \Phi) | t \Rightarrow_m^* \omega\},$$
$$\underline{LR}_m(G,t) = \{\omega \in T(\Sigma \cup X \cup \Phi) | t \Longleftrightarrow_m^* \omega\}.$$ □

We are now ready to define the nonterminal separating property for $m$-macro grammars.

**Definition 3.1.2.** An $m$-macro grammar $G = (\Phi,\Sigma,X,P,Z)$ has the *NTS property* or is an *NTS $m$-macro grammar* if for all $n \geq 0$, $A \in \Phi_n$, $\{x_1,...,x_n\} \subseteq X$,

$$\underline{LR}_m(G,A(\vec{x})) = \underline{L}_m(G,A(\vec{x})).$$ □

Here we consider the variables $x_1,...,x_n$ as members of a terminal alphabet $\Sigma'$ with $\Sigma \subseteq \Sigma'$, according to Fischer [6]; cf. also [4].

**Proposition 3.1.3.** *Let $G = (\Phi,\Sigma,X,P,Z)$ be an NTS $m$-macro grammar. Then for all $n,k \geq 0$, $A \in \Phi_n$, $B \in \Phi_k$, $\{x_1,...,x_n\} \subseteq X$ , $\{x_1,...,x_k\} \subseteq X$,*

$$\underline{L}_m(G,A(\vec{x}_{(n)})) \cap \underline{L}_m(G,B(\vec{x}_{(k)})) = \varnothing$$

*or*

$$\underline{L}_m(G,A(\vec{x}_{(n)})) = \underline{L}_m(G,B(\vec{x}_{(k)})).$$

*Proof:* Let $\omega$ be an element of $\underline{L}_m(G,A(\vec{x}_{(n)})) \cap \underline{L}_m(G,B(\vec{x}_{(k)}))$. Then $A(\vec{x}_{(n)}) \Rightarrow_m^* \omega$ as well as $B(\vec{x}_{(k)}) \Rightarrow_m^* \omega$ holds. This implies $A(\vec{x}_{(n)}) \Longleftrightarrow_m^* B(\vec{x}_{(k)})$. With the NTS property of $G$ we get $A(\vec{x}_{(n)}) \Rightarrow_m^* B(\vec{x}_{(k)})$ and $B(\vec{x}_{(k)}) \Rightarrow_m^* A(\vec{x}_{(n)})$ which implies $\underline{L}_m(G,A(\vec{x}_{(n)})) = \underline{L}_m(G,B(\vec{x}_{(k)}))$. □

We see that NTS $m$-macro grammars have a similar ‘‘nonterminal separating property’’ as context-free grammars; cf. Proposition 2.2.2.

**Example 3.1.4.** Consider the linear basic macro grammar $G = (\Phi,\Sigma,X,P,Z)$ with $\Phi = \Phi_0 \cup \Phi_3$, $\Phi_0 = \{S\} = Z$, $\Phi_3 = \{A\}$, $X = \{x,y,z\}$, $\Sigma = \{a,b,c,[,],\#\}$, and $P$ consists of the productions

$$S \rightarrow A(\lambda,\lambda,\lambda)$$
$$A(x,y,z) \rightarrow A(ax,by,cz)$$
$$A(x,y,z) \rightarrow [x\#y\#z]$$

The language generated by $G$ is $L(G) = \{[a^n\#b^n\#c^n] | n \geq 0\}$, and $\underline{L}(G,S) = \{S\} \cup \{A(a^n,b^n,c^n) | n \geq 0\} \cup L(G)$. Because $A(a^n,b^n,c^n)$, $(n \geq 1)$ only reduces to terms $A(a^k,b^k,c^k)$ with $0 \leq k < n$, and $[a^n\#b^n\#c^n]$ only reduces to $A(a^n,b^n,c^n)$, we have $\underline{L}(G,S) = \underline{LR}(G,S)$. A similar argument for $A(x,y,z)$ yields $\underline{L}(G,A(x,y,z)) = \underline{LR}(G,A(x,y,z))$; so $G$ is an NTS macro grammar. □

We see also that in case $\Phi = \Phi_0$ and, consequently, $G$ is a context-free grammar, Definition 3.1.2 corresponds to Definition 2.2.1 for context-free grammars.

### 3.2. Properties of NTS Macro Grammars

This section is devoted to some results which generalize Theorem 2.2.3 to $m$-macro grammars. To facilitate formulation and proofs we use the following notation.

**Definition 3.2.1.** Let $G=(\Phi,\Sigma,X,P,Z)$ be an $m$-macro grammar. Then $G$ has *property* $\Pi(m)$ if for all $A\in\Phi_n$, $B\in\Phi_k$, $u,\alpha u\beta\in T(\Sigma\cup X\cup\Phi)$, with $\{x_1,...,x_n\}\subseteq X$ and $\vec{\sigma}_{(k)}\in T^k(\Sigma\cup X\cup\Phi)$ the following implication holds

$$\text{if } A(\vec{x}_{(n)})\Rightarrow_m^*\alpha u\beta \text{ and } B(\vec{\sigma}_{(k)})\Rightarrow_m^* u,$$
$$\text{then } A(\vec{x}_{(n)})\Rightarrow_m^*\alpha B(\vec{\sigma}_{(k)})\beta. \qquad \square$$

First, we note that property $\Pi(m)$ is a natural extension of the property mentioned in Theorem 2.2.3 in the sense that if $\Phi=\Phi_0$, i.e., $G$ is context-free, the two properties coincide. To establish Theorem 3.2.3 we need the following lemma.

**Lemma 3.2.2.** *Let* $G$ *be an admissible* $m$*-macro grammar. Let* $\omega,\psi\in T(\Sigma\cup X\cup\Phi)$. *Then* $\omega\Rightarrow_{UNR}\psi$ *implies* $\omega\Longleftrightarrow_{OI}^*\psi$ *as well as* $\omega\Longleftrightarrow_{IO}^*\psi$. *As a corollary we have* $\omega\Rightarrow_{UNR}^*\psi$ *implies* $\omega\Longleftrightarrow_m^*\psi$ *for both* $m=OI$ *and* $IO$.

*Proof:* Let $\omega=\alpha A(\vec{\sigma})\beta$ with $A\in\Phi_n$, $n\geq 0$, $\vec{\sigma}\in T^n(\Sigma\cup X\cup\Phi)$ and $\psi=\alpha\delta(\vec{\sigma})\beta$. Then $\omega\Rightarrow_{UNR}\psi$ using the rule $A(\vec{x})\rightarrow\delta(\vec{x})$, $\delta(\vec{x})\in T(\Sigma\cup X\cup\Phi)$.
$m=OI$. First we have $\alpha A(\vec{\sigma})\beta\Rightarrow_{OI}^*\alpha' A(\vec{\sigma})\beta'$. This is the string obtained from $\omega$ such that every $A(\vec{\sigma})$ is on top level. Next we derive $\alpha' A(\vec{\sigma})\beta'\Rightarrow_{OI}^*\alpha'\delta(\vec{\sigma})\beta'$. Now all new occurrences of $\delta(\vec{\sigma})$ are on top level; so we can write $\alpha'\delta(\vec{\sigma})\beta'\Leftarrow_{OI}^*\alpha\delta(\vec{\sigma})\beta$.
$m=IO$. Similarly, using $A(\vec{\sigma})\Rightarrow_{IO}^* A(\vec{t})$, $A(\vec{t})\Rightarrow_{IO}^*\delta(\vec{t})$ and $\delta(\vec{t})\Leftarrow_{IO}^*\delta(\vec{\sigma})$, where $\vec{t}\in(\Sigma^*)^n$. $\square$

**Theorem 3.2.3.** *Let* $G$ *be an admissible* $m$*-macro grammar. Then* $(G,m)$ *is an NTS* $m$*-macro grammar if and only if* $G$ *has property* $\Pi(m)$.

*Proof:* First we prove the *if*-part. We have to show for $G$ satisfying $\Pi(m)$ that for each $A\in\Phi_n$ $(n\geq 0)$,

$$\underline{L}_m(G,A(\vec{x}))=\underline{LR}_m(G,A(\vec{x})).$$

The inclusion from left to right ($\subseteq$) is trivial. To establish the converse inclusion ($\supseteq$), we ought to prove that $A(\vec{x})\Longleftrightarrow_m^* t$ implies $A(\vec{x})\Rightarrow_m^* t$. This is done by induction on the length of $\Longleftrightarrow_m^*$.
Basic step ($p=0$): $A(\vec{x})\Longleftrightarrow_m^0 t$ implies $A(\vec{x})\Rightarrow_m^* t$ trivially.
Induction step. As induction hypothesis we take: $A(\vec{x})\Longleftrightarrow_m^p t$ implies $A(\vec{x})\Rightarrow_m^* t$.
Consider $A(\vec{x})\Longleftrightarrow_m^{p+1} t$. We distinguish two cases:

*Case 1.* $A(\vec{x})\Longleftrightarrow_m^p t'\Rightarrow_m t$. Obvious.

*Case 2.* $A(\vec{x}_{(n)})\Longleftrightarrow_m^p t'\Leftarrow_m t$. Suppose $t\Rightarrow_m t'$ by the derivation step $B(\vec{\sigma}_{(k)})\Rightarrow_m u$ and let $t=\alpha B(\vec{\sigma}_{(k)})\beta$, $t'=\alpha u\beta$ with $\alpha u\beta$, $u$, $B(\vec{\sigma}_{(k)})\in T(\Sigma\cup X\cup\Phi)$. By the induction hypothesis we have $A(\vec{x}_{(n)})\Rightarrow_m^* t'$. Using $\Pi(m)$ on $A(\vec{x}_{(n)})\Rightarrow_m^*\alpha u\beta$ and $B(\vec{\sigma}_{(k)})\Rightarrow_m u$ we get $A(\vec{x}_{(n)})\Rightarrow_m^*\alpha B(\vec{\sigma}_{(k)})\beta=t$. This completes the induction and the proof of the second inclusion.

To prove the *only if*-part we need the following. Let $G$ be an NTS $m$-macro grammar. Then for all $u$, $\alpha u\beta\in T(\Sigma\cup X\cup\Phi)$, $B\in\Phi_k$, $\vec{\sigma}_{(k)}\in T^k(\Sigma\cup X\cup\Phi)$,

$$B(\vec{\sigma}_{(k)})\Rightarrow_m^* u \text{ implies } \alpha B(\vec{\sigma}_{(k)})\beta\Longleftrightarrow_m^*\alpha u\beta.$$

It is easy to see that for $m=IO$ and $m=UNR$ this holds even without $G$ being NTS and with $\Rightarrow_m^*$ instead of $\Longleftrightarrow_m^*$. For $m=OI$ we obtain this implication as follows. If $B(\vec{\sigma}_{(k)})\Rightarrow_{OI}^* u$, then $B(\vec{\sigma}_{(k)})\Rightarrow_{UNR}^* u$ trivially; so $\alpha B(\vec{\sigma}_{(k)})\beta\Rightarrow_{UNR}^*\alpha u\beta$ and by Lemma 3.2.2. we have $\alpha B(\vec{\sigma}_{(k)})\beta\Longleftrightarrow_{OI}^*\alpha u\beta$. (Note that because $G$ is NTS, we now can even prove the stronger fact: $B(\vec{\sigma}_{(k)})\Rightarrow_{OI}^* u$ implies $\alpha B(\vec{\sigma}_{(k)})\beta\Rightarrow_{OI}^*\alpha u\beta$).

Now, if $A(\vec{x}_{(n)}) \Rightarrow^*_m \alpha u \beta$ and $B(\vec{\sigma}_{(k)}) \Rightarrow^*_m u$, then we get $A(\vec{x}_{(n)}) \Longleftrightarrow^*_m \alpha B(\vec{\sigma}_{(k)})\beta$. Since $(G,m)$ is NTS, we conclude with $A(\vec{x}_{(n)}) \Rightarrow^*_m \alpha B(\vec{\sigma}_{(k)})\beta$. □

### 3.3. The Pre-NTS Property for Macro Grammars

Closely connected to the NTS property for context-free grammars is the pre-NTS property [3,5,9]; informally, the pre-NTS property equals the NTS property formulated for terminal strings only. It is still an open problem whether these two properties are equivalent for context-free grammars [3,5,9].

In this section we introduce and study the pre-NTS property for $m$-macro grammars.

**Definition 3.3.1.** Let $G=(\Phi,\Sigma,X,P,Z)$ be an $m$-macro grammar with $Z\subseteq\Phi_0$. Then $(G,m)$ is *pre-NTS* or *has the pre-NTS property* if for all $A\in\Phi_n$ $(n\geq 0)$, and $\{x_1,\ldots,x_n\}\subseteq X$, $L_m(G,A(\vec{x}))=LR_m(G,A(\vec{x}))$ where $LR_m(G,A(\vec{x}))=\underline{LR}_m(G,A(\vec{x}))\cap(\Sigma\cup X)^*$. □

**Definition 3.3.2.** Let $G=(\Phi,\Sigma,X,P,Z)$ be an $m$-macro grammar with $Z\subseteq\Phi_0$. Then $G$ has *property* $\pi(m)$ if for all $A\in\Phi_n$ $(n\geq 0)$, $B\in\Phi_k$, $u',\alpha u\beta\in(\Sigma\cup X)^*$, $\{x_1,\ldots,x_n\}\subseteq X$, and $\vec{\tau}\in T^k(\Sigma\cup X\cup\Phi)$, the following implication holds:

$$\text{if } A(\vec{x})\Rightarrow^*_m \alpha u\beta,\ B(\vec{\tau})\Rightarrow^*_m u \text{ and } B(\vec{\tau})\Rightarrow^*_m u',$$
$$\text{then } A(\vec{x})\Rightarrow^*_m \alpha u'\beta.$$ □

We want to prove the equivalence of Definition 3.3.1 and Definition 3.3.2. It turns out to be the easiest way to do this by introducing a second property $\rho(m)$ which is equivalent to both of them.

**Definition 3.3.3.** An $m$-macro grammar $G$ has property $\rho(m)$ if for all $A\in\Phi_n$ $(n\geq 0)$, and $\{x_1,\ldots,x_n\}\subseteq X$, $t\in T(\Sigma\cup X\cup\Phi)$, $u,u'\in(\Sigma\cup X)^*$ the following implication holds:

$$\text{if } A(\vec{x})\Rightarrow^*_m u,\ t\Rightarrow^*_m u, \text{ and } t\Rightarrow^*_m u', \text{ then } A(\vec{x})\Rightarrow^*_m u'.$$ □

**Theorem 3.3.4.** *Let $G$ be an admissible $m$-macro grammar. Then the following statements are equivalent:*

(1) *$(G,m)$ is pre-NTS,*
(2) *$G$ has property $\pi(m)$,*
(3) *$G$ has property $\rho(m)$.*

*Proof:* (1)⇒(2): Suppose there exist derivations $B(\vec{\tau})\Rightarrow^*_m u$, $B(\vec{\tau})\Rightarrow^*_m u'$ and $A(\vec{x})\Rightarrow^*_m \alpha u\beta$ for $u'$, $\alpha u\beta\in(\Sigma\cup X)^*$. Because $\alpha u\beta$ is a word over $\Sigma\cup X$ there is no distinction between the three modes of reduction from $\alpha u\beta$. Therefore we have $A(\vec{x})\Rightarrow^*_m \alpha u\beta \Leftarrow^*_m \alpha B(\vec{\tau})\beta$. Now in $\alpha B(\vec{\tau})\beta$, $B(\vec{\tau})$ is on top level, so we continue with $\alpha B(\vec{\tau})\beta\Rightarrow^*_m \alpha u'\beta$ which is a word over $\Sigma\cup X$. Thus $A(\vec{x})\Longleftrightarrow^*_m \alpha u'\beta$ and, as $(G,m)$ is pre-NTS, $A(\vec{x})\Rightarrow^*_m \alpha u'\beta$. Hence $G$ has property $\pi(m)$.

(2)⇒(3): Let $A(\vec{x})\Rightarrow^*_m u$, $t\Rightarrow^*_m u$ and $t\Rightarrow^*_m u'$. Obviously, it is possible to write $t$ as an unique sequence of terms, viz. $t=t_1\ldots t_k$, such that no $t_i$ is a concatenation of two or more terms. It is clear that in expanding some $t_i$, none of the other terms $t_j$ is affected. So we can write $u$ as $u_1\ldots u_k$ and $u'$ as $u_1'\ldots u_k'$ with $t_i\Rightarrow^*_m u_i$ and $t_i\Rightarrow^*_m u_i'$, respectively. Now we have for some $i$, $1\leq i\leq k$ $A(\vec{x})\Rightarrow^*_m u_1\ldots u_i\ldots u_k$, $t_i\Rightarrow^*_m u_i$, $t_i\Rightarrow^*_m u_i'$, and with $\pi(m)$ we get $A(\vec{x})\Rightarrow^*_m u_1\ldots u_i'\ldots u_k$. We apply this argument to each $u_i$ consecutively, which finally yields $A(\vec{x})\Rightarrow^*_m u_1'\ldots u_k'=u'$ which is the desired result.

(3)⇒(1): We have to show $LR_m(G,A(\vec{x}))\subseteq L_m(G,A(\vec{x}))$, which we do by induction on the number of reduction steps in $A(\vec{x})\Longleftrightarrow^*_m w$, with $w\in(\Sigma\cup X)^*$. We denote this by $\Longleftrightarrow^{*n}_m$ which means that $\alpha\Longleftrightarrow^{*n}_m\beta$ holds if and only if $\alpha\Longleftrightarrow^*_m\beta$ in which $n$ reduction steps have been used.
Basic step $(n=0)$. $A(\vec{x})\Longleftrightarrow^{*0}_m w$ directly implies $A(\vec{x})\Rightarrow^*_m w$.

Induction step. As induction hypothesis we have: $A(\vec{x}) \Longleftrightarrow_m^{*n} w$ implies $A(\vec{x}) \Rightarrow_m^* w$. Let $A(\vec{x}) \Longleftrightarrow_m^{*n+1} w$. To show that $A(\vec{x}) \Rightarrow_m^* w$ we look at the last reduction step in $A(\vec{x}) \Longleftrightarrow_m^{*n+1} w$. We write this as $A(\vec{x}) \Longleftrightarrow_m^{*n} t \Leftarrow_m t' \Rightarrow_m^* w$. Because $G$ is admissible there is a word $u \in (\Sigma \cup X)^*$ with $t \Rightarrow_m^* u$. Applying the induction hypothesis we get $A(\vec{x}) \Rightarrow_m^* u$, with $t' \Rightarrow_m^* u$, and $t' \Rightarrow_m^* w$ and property $\rho(m)$ this gives us $A(\vec{x}) \Rightarrow_m^* w$. □

## 4. Concluding Remarks

In the previous section we generalized some characterizations of NTS and pre-NTS context-free grammars to corresponding statements for (pre-) NTS $m$-macro grammars. On the other hand one wants results that are specific for NTS macro grammars in the sense that there is no analogue for context-free grammars. Or, in other words, results that are due to the fact that we deal with macro grammars rather than context-free grammars.

A first example of such a result shows that NTS ‘‘reduced macro grammars’’, i.e., admissible NTS macro grammars with no initial symbols in the right-hand sides of their productions, are argument-preserving.

**Proposition 4.1.** *Let $G = (\Phi, \Sigma, X, P, Z)$ be an admissible NTS $m$-macro grammar, with no elements of $Z$ occurring in the right-hand side of any production. Then $G$ is argument-preserving.*

*Proof:* Suppose we have a production rule $A(x_1, \ldots, x_n) \rightarrow t$ with $A \notin \Phi_0$, which is not argument-preserving, say $x_i$ does not occur in $t$, $1 \leq i \leq n$. Suppose further that we have obtained a word $\omega \in T(\Sigma \cup \Phi)$ derived from some $S \in Z$ on which this rule is applicable. Writing $\omega$ as $\alpha A(\sigma_1, \ldots, \sigma_n)\beta$ we derive

$$\alpha t[\sigma_1/x_1, \ldots, \sigma_{i-1}/x_{i-1}, \sigma_{i+1}/x_{i+1}, \ldots, \sigma_n/x_n]\beta.$$

This last term however is, for instance, for some $T$ in $Z$ reducible to $\alpha A(\sigma_1, \ldots, \sigma_{i-1}, T, \sigma_{i+1}, \ldots, \sigma_n)\beta$, which we write as $\omega(T)$. So we have $S \Longleftrightarrow_m^* \omega(T)$. Since $G$ is NTS, we obtain $S \Rightarrow_m^* \omega(T)$. But no production rule can ever introduce a $T$ from $Z$ in a sentential form. Thus we cannot derive such a term $\omega(T)$ from $S$. □

The following statement is much more interesting. However, we are unable to prove it and therefore we formulate it as

**Conjecture 4.2.** *Each admissible NTS IO-macro grammar generates a basic macro language.* □

The first easy step in proving this conjecture, consists of the following observation.

**Lemma 4.3.** *Let $G$ be an admissible NTS IO-macro grammar. Then for all $A \in \Phi$,*

$$\underline{L}_{UNR}(G, A(\vec{x})) = \underline{L}_{IO}(G, A(\vec{x})).$$

*Proof:* We only have to show $\underline{L}_{UNR}(G, A(\vec{x})) \subseteq \underline{L}_{IO}(G, A(\vec{x}))$, since the converse inclusion is trivial. Let $t \in T(\Sigma \cup X \cup \Phi)$ and $A(\vec{x}) \Rightarrow_{UNR}^* t$. Then we have by Lemma 3.2.2 $A(\vec{x}) \Longleftrightarrow_{IO}^* t$, and using the fact that $(G, IO)$ is NTS, we obtain $A(\vec{x}) \Rightarrow_{IO}^* t$. □

In order to complete the proof of Conjecture 4.2 it is sufficient to establish

**Conjecture 4.4.** *Let $G$ be an NTS IO-macro grammar that contains a nested production*

$$A(\vec{x}) \rightarrow B(\vec{\gamma}(\vec{x})) \qquad (*)$$

*i.e., some entry of $\vec{\gamma}$ contains a nonterminal symbol. If $\beta(\vec{x}) \in L_{UNR}(G, B(\vec{x}))$, then in the derivation $A(\vec{x}) \Rightarrow_{IO}^* \beta(\vec{\gamma}(\vec{x}))$ the rule $(*)$ has not been applied.* □

*Acknowledgment.* I thank Peter Asveld for his helpful comments and for his aid during the preparation of the text.

**References**

1. P.R.J. Asveld: Time and space complexity of inside-out macro languages, *Internat. J. Comput. Math.* **10** (1981) 3-14.

2. J.M. Autebert, L. Boasson & G. Sénizergues: Langages de parenthèses, langages N.T.S. et homomorphismes inverses, *R.A.I.R.O. Inform. théor./Theor. Inform.* **18** (1984) 327-344.

3. L. Boasson: Dérivations et réductions dans les grammaires algébriques, *in* ‘‘7th International Colloquium on Automata Languages and Programming’’ Lect. Notes Comp. Sci. **85** (1980) 109-118, Springer-Verlag, Berlin - Heidelberg - New York.

4. J. Engelfriet, E.M. Schmidt & J. van Leeuwen: Stack machines and classes of non-nested macro languages, *J. Assoc. Comput. Mach.* **27** (1980) 96-117.

5. L. Boasson & G. Sénizergues: NTS languages are deterministic and congruential, *J. Comput. System Sci.* **31** (1985) 332-342.

6. M.J. Fischer: *Grammars with Macro-like Productions*, Ph.D. Thesis (1968), Harvard University, Cambridge, Mass.

7. M.J. Fischer: Grammars with macro-like productions, Proc. 9th Ann. IEEE Symp. on Switching and Automata Theory (1968) 131-142.

8. Ch. Frougny: Simple deterministic NTS languages, *Inform. Process. Lett.* **12** (1981) 174-178.

9. G. Sénizergues: A new class of C.F.L. for which the equivalence is decidable, *Inform. Process. Lett.* **13** (1981) 30-34.

10. G. Sénizergues: The equivalence and inclusion problems for NTS languages, *J. Comput. System Sci.* **31** (1985) 303-331.