Acta Informatica 8, 193 – 199 (1977)

Acta Informatica

Van Wijngaarden Grammars and Space Complexity Class EXSPACE

P. Deussen [1] and K. Mehlhorn [2]

[1] Institut für Informatik I der Universität Karlsruhe, Postfach 6380, D-7500 Karlsruhe 1, Federal Republic of Germany *

[2] Fachbereich 10, Angewandte Mathematik und Informatik der Universität Saarbrücken, D-6600 Saarbrücken, Federal Republic of Germany

Summary. Top-down and bottom-up decision strategies for van Wijngaarden grammars led to type R and type L van Wijngaarden grammars. The corresponding language families are now shown to be equal and, furthermore, to equal EXSPACE. Thus, EXSPACE is characterised syntactically, and the closure properties of type L and type R languages are those of EXSPACE.

0. Introduction

Regarding the hyperrules of a van Wijngaarden grammar as a means for generating infinitely many contextfree productions by which from a startsymbol the terminal words of the generated language are produced, one is naturally led to the idea of top-down and bottom-up decision procedures for such grammars.

These decision procedures work for type R and for type L grammars respectively [1].

Let $L_L[L_R]$ denote the class of all languages generated by type L [type R] van Wijngaarden grammars. L_L contains all contextsensitive languages and, as we shall see, so does L_R.

The symmetry of reasoning in introducing both types of van Wijngaarden languages as well as a big lot of examples suggested the conjecture $L_L = L_R$. At the first attempt, however, this conjecture could not be proved.

The aim of this report is to close this gap. In the sequel, the inclusions

$$L_R \subset L_L \subset \text{EXSPACE} \subset L_R$$

will be shown in that order where EXSPACE is the class of languages whose space complexity is c^n. The first and the last inclusion exhibits characteristic techniques in handling van Wijngaarden grammars, the second inclusion consists in a estimation of numerical bounds.

* Address for offprint requests

Thus, $L_L = L_R$ on one hand and $L_L = L_R = \text{EXSPACE}$ (=NEXSPACE, see [2]) on the other: as a byproduct, EXSPACE is characterised independently in a syntactic way by type R (or type L) grammars which fact seems to be of interest in its own right. The result furthermore states that L_R properly contains the class CS of contextsensitive languages and, in terms of tape complexity, is much larger than CS. Closure properties of L_L, L_R are then characterised by those of EXSPACE.

To avoid lengthy repeats the reader is assumed to be familiar with the content of [1].

1. $L_R \subset L_L$

As in the theory of Chomsky grammars, one defines right- and leftlinear hyperrules:

Definition. A van Wijngaarden grammar is called *right-* [*left-*]*linear* if all its hyperrules are of either of the forms:

$$\langle \bar{u} \rangle ::= x \langle \bar{u}_1 \rangle \qquad [\langle \bar{u} \rangle ::= \langle \bar{u}_1 \rangle x]$$
$$\langle \bar{u} \rangle ::= x$$
$$(\bar{u}, \bar{u}_1 \in (\mathscr{M} \cup \mathscr{T})^+, \qquad x \in \Sigma^*).$$

Without loss of generality we may assume that the startsymbol $\langle s \rangle$ is shown and does not appear on the right hand side of any hyperrule.

Then by the following translation table

$$\langle s \rangle ::= x \qquad \bullet\!\!-\!\!\bullet \; \langle s \rangle \;::= x$$
$$\langle s \rangle ::= x \langle \bar{u}_1 \rangle \; \bullet\!\!-\!\!\bullet \; \langle \bar{u}_1 \rangle ::= x$$
$$\langle \bar{u} \rangle ::= x \langle \bar{u}_1 \rangle \; \bullet\!\!-\!\!\bullet \; \langle \bar{u}_1 \rangle ::= \langle \bar{u} \rangle x$$
$$\langle \bar{u} \rangle ::= x \qquad \bullet\!\!-\!\!\bullet \; \langle s \rangle \;::= \langle \bar{u} \rangle x$$

rightlinear van Wijngaarden grammars are converted to equivalent leftlinear ones, and vice versa. The idea is the same as for Chomsky-3 grammars. Lemma 2 in [1] shows that to each van Wijngaarden grammar there exists a leftlinear van Wijngaarden grammar equivalent to the former.

If a right- [left-]linear van Wijngaarden grammar is of type R (or of type L) then the left- [right-]linear grammar as obtained by the above translation is of type L (of type R).

Consequently, the following theorem concludes part 1.

Theorem. To each type R van Wijngaarden grammar there is an equivalent rightlinear grammar of type R [4].

Proof. Due to Lemma 1 in [1] we may assume that the van Wijngaarden grammar $\mathscr{W}$ is in separated form, and its terminating rules are

$$\langle h_i \rangle ::= \alpha_i \qquad i = 1, \ldots, n \qquad (\alpha_i \in \Sigma).$$

Construction of the rightlinear grammar $\mathscr{W}'$:

1. $\mathscr{T}' := \mathscr{T} \cup \{:, +\}$,

$\mathscr{M}' := \mathscr{M} \cup \{H, K, M, N\}$.

All added symbols are assumed to be completely new.

2. To the metaproductions of $\mathscr{W}$ add productions to obtain

$\mathscr{L}_H = \mathscr{L}_K = \{h_1, h_2, \ldots, h_n, +\}^*$,

$\mathscr{L}_M = \mathscr{L}_N = (\mathscr{T} \cup \{+\})^*$.

3. The hyperrules of $\mathscr{W}$ are transformed as follows:

$\bar{\Gamma}$	$\bar{\Gamma}'$
1. $\langle \bar{u} \rangle ::= \langle \bar{u}_1 \rangle$	1'. $\langle H : M + \bar{u} + N \rangle ::= \langle H : M + \bar{u}_1 + N \rangle$
2. $\langle \bar{u} \rangle ::= \langle \bar{u}_1 \rangle \langle \bar{u}_2 \rangle \ldots \langle \bar{u}_k \rangle$ $(k \geqq 2)$	2'. $\langle H : M + \bar{u} + N \rangle ::= \alpha_i \langle H\, h_i + : M + \bar{u}_1 + \bar{u}_2 + \cdots + \bar{u}_n + N \rangle$ for $i = 1 \ldots n$
	3'. $\langle H + : H + K \rangle ::= \langle + K \rangle$
	4'. $\langle + h_i + K \rangle ::= \alpha_i \langle + K \rangle$
5. $\langle h_i \rangle ::= \alpha_i$	5'. $\langle + h_i + \rangle ::= \alpha_i \quad (i = 1 \ldots n)$

The startsymbol of $\mathscr{W}'$ is $\langle + : + s + \rangle$.

As already mentioned, we may assume that the hyperrules of $\mathscr{W}$ show the startsymbol $\langle s \rangle$ and $\langle s \rangle$ does not occur on any right hand side. Therefore the same holds for $\bar{\Gamma}'$ which is, by construction, rightlinear. $\bar{\Gamma}$ is of type R by assumption and, hence, so is $\bar{\Gamma}'$.

The transformation is similar to that used in Lemma 2 in [1], the difference being that in hyperrules 2'. terminals α_i are generated nondeterministically and their corresponding h_i's are kept in H. This is done to force that the transformed rules 2'. need be rightbound only and not strictly right-bound.

$L(\mathscr{W}) \subset L(\mathscr{W}')$:

Since $\bar{\Gamma}$ is separated, any derivation $\langle s \rangle \underset{\bar{\Gamma}}{\Rightarrow} x$ can be rearranged to read

$$\langle s \rangle \Rightarrow \langle h_i \rangle \langle h_j \rangle \cdots \langle h_s \rangle \langle h_r \rangle \cdots \langle h_k \rangle \Rightarrow x$$

where in the first part rules 1. and 2. are applied, and in the second part only rules 5. By the above transformation one obtains from the first derivation part:

$$\langle + : + s + \rangle \Rightarrow$$
$$\alpha_1\, \alpha_m \ldots \alpha_n \cdot \langle + h_1 + h_m + \cdots + h_n + : + h_i + h_j + \cdots + h_s + h_r + \cdots + h_k + \rangle.$$

It is important to note that in any case $|\alpha_l\, \alpha_m \ldots \alpha_n|$ is less than the number of h's in $+ h_i + h_j + \cdots + h_s + h_r + \cdots + h_k +$ since, in $\bar{\Gamma}$, rules 2. are applied at most $|x| - 1$

times. Therefore rule 3'. is applicable if we have chosen the $\alpha_l \alpha_m \ldots \alpha_n$ to be the first symbols of x:

$$\to \alpha_i \alpha_j \ldots \alpha_s \langle + h_r + \cdots + h_k + \rangle.$$

Then, by rules 4'. and 5'., this can be derived into x. $L(\mathscr{W}') \subset L(\mathscr{W})$ is now obvious from the mechanism of $\bar{\Gamma}'$. ▮

Except for rule 3'. the transformation used in the proof would also work for type L grammars, but the exceptional rule is essential and at the same time essentially not (strictly) leftbound.

2. $L_L \subset$ EXSPACE

Let $A \in L_L$ be a language and let $\mathscr{W}$ be a type L grammar for A. We show how to simulate the derivations of $\mathscr{W}$ in nondeterministic exponential space.

It should be obvious that the crucial step is to derive a bound on the length of intermediate sentential forms in a derivation. We show that their length is bounded by an exponential function of the length of the string derived.

The hyperrules of $\mathscr{W}$ are of the form

$$\langle \bar{u} \rangle ::= \sigma_1 \langle \bar{u}_1 \rangle \sigma_2 \langle \bar{u}_2 \rangle \ldots \langle \bar{u}_k \rangle \sigma_{k+1}.$$

Since $\mathscr{W}$ is a type L grammar there exists an integer c such that

$$|\bar{u}|_M \leqq c\,|\bar{u}_1 \ldots \bar{u}_k|_M \quad \text{for all } M \in \mathscr{M}$$

and

$$|u|_{\mathscr{T}_0} \leqq c$$

for all hyperrules of $\mathscr{W}$.

We split the set of hyperrules $\bar{\Gamma}$ of $\mathscr{W}$ into two disjoint sets $\bar{\Gamma}_1$ and $\bar{\Gamma}_2 = \bar{\Gamma} - \bar{\Gamma}_1$. $\bar{\Gamma}_1$ consists of all hyperrules of the form $\langle \bar{u} \rangle ::= \langle \bar{u}_1 \rangle$. The decomposition of the set of rules induces a decomposition of the derivations $\langle s \rangle \Rightarrow x \in \Sigma^*$, namely

$$\langle s \rangle = v_0 \underset{\bar{\Gamma}_1}{\Longrightarrow} w_1 \underset{\bar{\Gamma}_2}{\longrightarrow} v_1 \underset{\bar{\Gamma}_1}{\Longrightarrow} w_2 \underset{\bar{\Gamma}_2}{\longrightarrow} v_2 \underset{\bar{\Gamma}_1}{\Longrightarrow} \cdots \underset{\bar{\Gamma}_2}{\longrightarrow} v_{n-1} \underset{\bar{\Gamma}_1}{\Longrightarrow} w_n \underset{\bar{\Gamma}_2}{\longrightarrow} v_n = x$$

where $w_i, v_i \in (\Phi \cup \Sigma)^*$. We prove several simple facts about the length of the strings w_i and v_i with respect to the alphabet $\mathscr{T}_0 \cup \Sigma$.

From strict left-boundness of all hyperrules in $\bar{\Gamma}_1$, we obtain

$$|v_i|_{\mathscr{T}_0 \cup \Sigma} \leqq |w_{i+1}|_{\mathscr{T}_0 \cup \Sigma} \quad (0 \leqq i \leqq n-1). \tag{1}$$

v_i is derived from w_i by applying a single production of Γ_2 for which

$$|u|_{\mathscr{T}_0} \leqq c + c\,|u_1 u_2 \ldots u_k|_{\mathscr{T}_0}$$

holds due to the properties of the constant c. Consequently we get

$$|w_i| \leqq c(|v_i| + 1),$$

and, using (1), by induction

$$|w_{n-i}| \leqq c^{i+1} |v_n| + \sum_{j=1}^{i+1} c^j.$$

This last inequality together with (1) and $n \leqq 2|x| - 1$ (see [1]) yields for $i = n-1$:

$$|v_i|, |w_i| \leqq c^{2|x|-1} |x| + \sum_{j=1}^{2|x|-1} c^j.$$

The length of intermediate sentential forms is thus bounded by an exponential in the length of the string derived. It is now easy (and left to the reader) to built a nondeterministic exponentially space bounded TM which accepts A. In view of Savitch's result (see [2]) NSPACE $(S_{()}) \subset$ SPACE $(S^2{}_{()})$ for all tape-constructable functions $S_{()} \geqq \log_{()}$ this suffices.

3. EXSPACE $\subset L_R$

3.1. General Considerations

The technique used to show this result bears some general aspects and symmetries that are worthwile to be discussed.

Consider a Semi-Thue system

$$\mathcal{S} = (V, \Pi)$$

with finite vocabulary V containing Σ, a distinguished subset, and with finite set Π of productions $p::=q$. In the usual way, Π gives rise to the relation $\rightarrow$ of direct derivation, and to its reflexive, transitive closure $\Rightarrow$.

For $\mathcal{S}$ and some $s \in V^*$ define

$$L(\mathcal{S}, s) = \{x \in \Sigma^* : s \Rightarrow x\},$$

the set of terminal words generated by $\mathcal{S}$ from s and, dually,

$$A(\mathcal{S}, s) = \{x \in \Sigma^* : x \Rightarrow s\},$$

the set of terminal words accepted by $\mathcal{S}$ from s. (Actually, it is this duality that goes through the whole theory of formal languages like a read thread, although it is in most cases obscured by an inappropriate terminology.)

If Π consists of Chomsky-productions and $s \in V \setminus \Sigma$ is the startsymbol then $L(\mathcal{S}, s)$ is a Chomsky language. Taking then Π^c to consist of the converse productions (i.e. $q::=p$), we obtain $L(\mathcal{S}, s) = A(\mathcal{S}^c, s)$, a simple acceptor. Later on we shall slightly change such acceptors in order to include the usual finite control by states.

Van Wijngaarden grammars now allow for incorporating both the generating and the accepting aspect into one grammar. This is *the* requisit to generate the accepted set $A(\mathcal{S}, s)$ by a van Wijngaarden grammar without changing Π!

The following tables show how van Wijngaarden grammars generating $L(\mathscr{S}, s)$ and $A(\mathscr{S}, s)$ may be constructed.

Metasystem:

Metanotions: T with $\mathscr{L}_T = \Sigma$,
L, R, W with $\mathscr{L}_T = \mathscr{L}_R = \mathscr{L}_W = V^*$.

Hyperrules:

$L(\mathscr{S}, s)$	$A(\mathscr{S}, s)$
start: $\langle s \rangle$	$\langle : t_0\, t_0 \rangle$ for some arbitrary but fixed $t_0 \in \Sigma$
	1'. $\langle : t_0\, t_0 \rangle \ ::= \langle : t\, t \rangle$ for all $t \in \Sigma$ 2'. $\langle W : TT \rangle ::= t \langle Wt : TT \rangle$ for all $t \in \Sigma$ 3'. $\langle W : TT \rangle ::= \langle WT : T \rangle$
4. $\langle LpR \rangle ::= \langle LqR \rangle$	4'. $\langle LpR : T \rangle ::= \langle LqR : T \rangle$
for all $p ::= q \in \Pi$	
5. $\langle Wt \rangle \ ::= \langle W \rangle t$	5'. $\langle s : t \rangle \ ::= t$
for all $t \in \Sigma$	

The left column uses the idea of [3, 1]. The right column was found by [4]: rule 2'. generates the terminal string highly nondeterministically and keeps track of it in $\langle Wt : TT \rangle$; 3'. then switches to the accepting rules 4'. by adding the "last" terminal to W without loosing it: $\langle WT : T \rangle$; 4'. leads to $\langle s : t \rangle$ if and only if the terminal string just generated plus this t belongs to $A(\mathscr{S}, s)$.

In both columns, the type of the van Wijngaarden grammar – type R or L – depends solely on Π: rules 1'., 3'. are strongly left – and right – bound, and rules 2'., 5., 5'. are left – and rightbound. Because of rules 4., 4'. both grammars are of type R if and only if $|p| \geqq |q|$ for all $p ::= q \in \Pi$ and, vice versa, for type L.

As a preliminary result we see therefore that each contextsensitive language is in L_L as well as in L_R: take $s \in V \setminus \Sigma$ and let $|p| \leqq |q|$ for each $p ::= q \in \Pi$, then $L(\mathscr{S}, s)$ is contextsensitive, and the left column is of type L; consequently, for each $q ::= p \in \Pi^c$, $|q| \geqq |p|$ holds. And, using Π^c, the right column will be of type R, furthermore $L(\mathscr{S}, s) = A(\mathscr{S}^c, s)$.

3.2. *Proof of* $EXSPACE \subset L_R$

The right column in the above table provides the tool for proving our last result.

A (one sided) Turing-acceptor with tape-complexity c^n can be regarded as the following Semi-Thue system:

$V = V_0 \cup \Sigma \cup \{\not{b}\} \cup Q$.

Here, $\flat$ is the 'blank'-symbol, and Q is a (finite) set of 'states' containing the set F of final states. $q_0 \in Q$ is the initial state.

Π

consists of productions of the following forms only:

$$\left.\begin{array}{l} q\,v ::= v'\,q' \\ \forall\, l \in V_0 \cup \Sigma : l\,q\,v ::= q'\,l\,v' \end{array}\right. \quad \text{where} \begin{cases} v \in V_0 \cup \Sigma \cup \{\flat\} \\ v' \in V_0 \cup \Sigma \\ q, q' \in Q. \end{cases}$$

The accepted set then is

$$A = \{x \in \Sigma^* : \exists\, r, s \in (V_0 \cup \Sigma \cup \{\flat\})^* \; \exists\, q \in F : q_0\, x\, b^{(c|x| - |x|)} \Rightarrow r\, q\, s\}.$$

The corresponding van Wijngaarden grammar is:

Metasystem

T, L, R, W as in Section 3.1

B with $\mathscr{L}_B = \{\flat\}^*$.

Hyperrules

start:	$\langle q_0\, \flat^c : t_0\, t_0 \rangle$	for some $t_0 \in \Sigma$
1'.	$\langle q_0\, \flat^c : t_0\, t_0 \rangle ::= \langle q_0\, \flat^c : t\, t \rangle$	for all $t \in \Sigma$
2'.	$\langle q_0\, WB : TT \rangle ::= t \langle q_0\, Wt\, B^c : TT \rangle$	
3'.	$\langle q_0\, WB : TT \rangle ::= \langle q_0\, WTB : T \rangle$	
4'.	$\langle L a R : T \rangle ::= \langle L d R : T \rangle$	for each $a ::= d \in \Pi$
5'.	$\langle L q R : t \rangle ::= t$	for each $q \in F,\ t \in \Sigma$

Hyperrules 1'., 3'. are strictly rightbound, and 4'. also because of Π; rule 2', 5'. are rightbound, which concludes the proof. ▮

Acknowledgements. Some of the initial ideas which led to Sections 1 and 3.1 are due to Eva Unvericht who firstly applied them in her diploma thesis [4]: we are grateful for this progress.

References

1. Deussen, P.: A decidability criterion for van Wijngaarden grammars. Acta Informatica **5**, 353–375 (1975)
2. Savitch, W.J.: Relationship between nondeterministic and deterministic tape complexities. J. Computer System Sci. **4**, 177–192 (1970)
3. Sintzoff, M.: Existence of a van Wijngaarden syntax for every recursively enumerable set. Annales de la Société Scientifique de Bruxelles, T. 81, II (1967)
4. Unvericht, E.: Van Wijngaarden Grammatiken vom Typ *L* und Typ *R*. Diplomarbeit am Institut für Informatik I, Universität Karlsruhe, 1976

Received October 18, 1976