Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

85

Automata, Languages and Programming

Seventh Colloquium, Noordwijkerhout, July 1980

Edited by

J. W. de Bakker and J. van Leeuwen

Springer-Verlag
Berlin Heidelberg New York

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

85

Automata, Languages and Programming

Seventh Colloquium
Noordwijkerhout, the Netherlands
July 14–18, 1980

EATCS

Edited by
J. W. de Bakker and J. van Leeuwen

Springer-Verlag
Berlin Heidelberg New York 1980

Editors

Jaco de Bakker
Mathematical Centre, Kruislaan 413
1098 SJ Amsterdam, the Netherlands

Jan van Leeuwen
Dept. of Computer Science, University of Utrecht
P.O.Box 80.002
3508 TA Utrecht, the Netherlands

AMS Subject Classifications (1979): 68-XX
CR Subject Classifications (1974): 4.1, 4.2, 5.2, 5.3

ISBN 3-540-10003-2 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-10003-2 Springer-Verlag New York Heidelberg Berlin

Printing and binding: Beltz Offsetdruck, Hemsbach/Bergstr.
2145/3140-543210

PREFACE

ICALP is the acronym of the annual International Colloquium on Automata, Languages and Programming sponsored by the European Association for Theoretical Computer Science (EATCS). It is a broad-based conference covering all aspects of the foundations of computer science, including such topics as automata theory, formal language theory, analysis of algorithms, computational complexity, computability theory, mathematical aspects of programming language definition, semantics of programming languages, program verification, theory of data structures and theory of data bases. Previously ICALP conferences were held in Paris (1972), Saarbrücken (1974), Edinburgh (1976), Turku (1977), Udine (1978) and in Graz (1979).

ICALP 80 is the 7th conference of EATCS, covering once again a broad spectrum of theoretical computer science. ICALP 80 was organized by the University of Utrecht and the Mathematical Centre at Amsterdam and was held July 14-18, 1980, in Noordwijkerhout, the Netherlands. The program committee consisted of J.W. de Bakker (Amsterdam, chairman), A. Blikle (Warsaw), C. Böhm (Rome), H.D. Ehrich (Dortmund), S. Even (Haifa), P. van Emde Boas (Amsterdam), I.M. Havel (Prague), J. van Leeuwen (Utrecht), H. Maurer (Graz), L.G.L.T. Meertens (Amsterdam), K. Mehlhorn (Saarbrücken), A.R. Meyer (MIT), R. Milner (Edinburgh), U. Montanari (Pisa), M. Nivat (Paris), M. Paterson (Coventry), G. Rozenberg (Leiden), A. Salomaa (Turku), J.W. Thatcher (Yorktown Heights), J. Vuillemin (Paris). We wish to thank the members of the program committee for their arduous job of evaluating the record number of 169 papers that were submitted to the conference. On their behalf we extend our gratitude to the referees which assisted this process (see next page).

ICALP 80 has been made possible by the support from a number of sources. We thank the Dutch Ministry for Education and Sciences (The Hague), the Mathematical Centre (Amsterdam), the University of Utrecht, the University of Leiden, CDC-the Netherlands and IBM-the Netherlands for sponsoring the conference. A special tribute goes to Mrs. S.J. Kuipers of the Mathematical Centre (Amsterdam) for her expert assistance in all organizational matters related to the conference.

We feel that ICALP 80 has succeeded in bringing together a variety of important developments in modern theoretical computer science. The need for a thorough investigation of the foundations of computer science evidently is increasing rapidly, as computer science moves on to ever more complex and diverse systems and applications. We hope that the ICALP conferences will continue to be an exponent of this trend in the years to come.

J.W. de Bakker and J. van Leeuwen

I C A L P 80

REFEREES FOR ICALP 80

H. Alblas
V. Ambriola
P. Ancilotti
K.R. Apt
A. Arnold
P.R.J. Asveld
G. Ausiello

R.J. Back
J. Bergstra
E. Börger
D.P. Bovet
F.J. Brandenburg
A. de Bruin
J.A. Brzozowski
R.M. Burstall

D. de Champeaux
M.P. Chytil
E.M. Clarke jr.
A.B. Cremers
K. Culik II

P. Degano
G. Degli Antoni
P. Della Vigna
P. Dembinski
M. Dezani- Ciancaglini
G. Dittrich

H. Edelsbrunner
E. Edens
J. Engelfriet

G. Filè
P. Flajolet
N. Francez

S. Gal
Z. Galil
H.J.M. Goeman
S.L. Graham
S.A. Greibach
G. Guiho
J.V. Guttag

D. Harel
T. Harju
M.A. Harrison

W. Imrich
A. Itai

J.M. Jaffe
T.M.V. Janssen
H. Janssens
K. Jensen
P. Jirků
N. Jones

J. Karhumäki
O. Kariv
H.C.M. Klein
S.R. Kosaraju
D.C. Kozen

J.K. Lenstra
G. Levi
S. Levialdi
M.R. Levy
M. Linna
E. Lipeck
M.H. Löb
G. Longo
R. Loos
B.E. Lub

A. Machi
A. Maggiolo-Schettini
J. Małuszyński
Z. Manna
A. Martelli
A. Mazurkiewicz
B. Mayoh
E. Meineche Schmidt
B. Melichar
W. Merzenich
C. Montangero
P.D. Mosses
K. Müller

M. Nielsen
A. Nijholt

Th. Ottmann
M.H. Overmars

P. Paolini
M. Penttonen
J. Pittl
G.D. Plotkin
A. Poigné
V.R. Pratt
G.A. Prini
H. Prodinger

M.O. Rabin
J.C. Reynolds
M.M. Richter
M. Rodeh
F. Romani
K. Ruohonen
P. Ružička

A. Salwicki
W.J. Savitch
W.L. Scherlis
G. Schlageter
A. Shamir
E. Shamir
Y. Shiloach
M. Sintzoff
M. Sipser
A. Skowron
M. Steinby

A. Tang
A. Tarlecki
J. Tiuryn
P.R. Torregiani
J.V. Tucker

F.J. Urbanek

M. Venturini-Zilli
R. Verraedt
P.M.B. Vitányi

R.W. v.d. Waall
W.W. Wadge
E.G. Wagner
M. Wand
L. Wegner
J. Winkowski

A. Yehudai

CONTENTS

W. Ainhirn
How to get rid of pseudoterminals 1

J. Albert and K. Culik II
Test sets for homomorphism equivalence on context free languages 12

J. Albert and L. Wegner
Languages with homomorphic replacements 19

H. Alt
Functions equivalent to integer multiplication 30

E. Astesiano and G. Costa
Languages with reducing reflexive types 38

R.-J. Back
Semantics of unbounded nondeterminism 51

R.I. Becker, Y. Perl and St.R. Schach
A shifting algorithm for min-max tree partitioning 64

J.A. Bergstra and J.V. Tucker
A characterisation of computable data types by means of a finite equational specification method 76

P. Berman
A note on sweeping automata 91

D. Bini
Border rank of a pxqx2 tensor and the optimal approximation of a pair of bilinear forms 98

L. Boasson
Derivations et réductions dans les grammaires algébriques 109

P. Cousot and R. Cousot
Semantic analysis of communicating sequential processes 119

E. Ehrenfeucht and G. Rozenberg
DOS systems and languages 134

H. Ehrig, H.-J. Kreowski and P. Padawitz
Algebraic implementation of abstract data types: concept, syntax, semantics and correctness 142

H. Ehrig, H.-J. Kreowski, J. Thatcher, E. Wagner and J. Wright
Parameterized data types in algebraic specification languages 157

E.A. Emerson and E.M. Clarke
Characterizing correctness properties of parallel programs using fixpoints . . 169

J. Engelfriet and G. Filè
Formal properties of one-visit and multi-pass attribute grammars 182

S. Even and Y. Yacobi
Cryptocomplexity and NP-completeness 195

Ph. Flajolet and J.-M. Steyaert
On the analysis of tree-matching algorithms 208

G.N. Frederickson and D.B. Johnson
Generating and searching sets induced by networks 221

M. Fürer
The complexity of the inequivalence problem for regular expressions with intersection 234

Z. Galil
An almost linear time algorithm for computing a dependency basis in a relational data base 246

H.J. Genrich and P.S. Thiagarajan
Bipolar synchronization systems 257

A. Goralcíková, P. Goralcík and V. Koubek
Testing of properties of finite algebras 273

J. Gray
A transaction model (Invited address) 282

M. Hennessy and R. Milner
On observing nondeterminism and concurrency 299

G. Hornung and P. Raulefs
Terminal algebra semantics and retractions for abstract data types 310

Th.-D. Huynh
The complexity of semilinear sets 324

J.R. Kennaway and C.A.R. Hoare
A theory of nondeterminism 338

D. Kozen
*A representation theorem for models of *-free PDL* 351

H. Langmaack and E.-R. Olderog
Present-day Hoare-like systems for programming languages with procedures: power, limits and most likely extensions 363

H.R. Lewis and Ch.H. Papadimitriou
Symmetric space-bounded computation 374

A. de Luca and A. Restivo
On some properties of local testability 385

M.E. Majster-Cederbaum
Semantics: algebras, fixed points, axioms 394

J.A. Makowsky
Measuring the expressive power of dynamic logics: an application of abstract model theory 409

K. Mehlhorn
Pebbling mountain ranges and its application to DCFL-recognition 422

E. Meineche Schmidt
Space-restricted attribute grammars 436

P. Mosses
A constructive approach to compiler correctness 449

Ch.H. Papadimitriou and J.L. Bentley
A worst-case analysis of nearest neighbor searching by projection 470

J.-E. Pin
Propriétés syntactiques du produit non ambigu 483

K.-J. Räihä and E. Ukkonen
On the optimal assignment of attributes to passes in multi-pass attribute evaluators 500

J.C. Raoult and J. Vuillemin
Optimal unbounded search strategies 512

R. Reischuk
A "fast implementation" of a multidimensional storage into a tree storage . . 531

A. Salomaa
Grammatical families (Invited address) 543

P. Schlichtiger
Partitioned chain grammars 555

J. Schmidt and E. Shamir
An improved program for constructing open hash tables 569

A. Shamir
On the power of commutativity in cryptography (Invited address) 582

S. Sippu and E. Soisalon-Soininen
Characterizations of the LL(k) property 596

M.B. Smyth

Computability in categories 609

M. Snir

On the size complexity of monotone formulas 621

T. Toffoli

Reversible computing 632

V.F. Turchin

The use of metasystem transition in theorem proving and program optimization . 645

P.M.B. Vitányi

On the power of real-time Turing machines under varying specifications 658

M.A. Frumkin, G.V. Gens, Ju.I. Hmelevskii and E.V. Levner

On reducibility among extremal combinatorial problems (paper not received)

HOW TO GET RID OF PSEUDOTERMINALS

W. Ainhirn
Institut für Informationsverarbeitung
Technische Universität Graz
Steyrergasse 17
A-8010 Graz/Austria

Abstract

We investigate the role of pseudoterminals for EOL forms. This leads us to the definition of m - interpretation which avoids pseudoterminals. We solve the problem of m - completeness of short and simple EPOL forms and finally consider the validity of some basic results on EOL forms under m - interpretation.

Introduction and Preliminaries

Investigations of EOL forms in [AiM], [CM], [CMO] and [MSW2] have shown in the past that we actually deal with three (rather than two!) alphabets: the terminal, the pseudoterminal and the nonterminal alphabet. The pseudoterminal alphabet contains those symbols which are explicitely specified as terminal symbols but never occur in the language generated by the system. It seems that pseudoterminals sometimes act rather pathologically. For this reason we define a new type of interpretation, so - called marvellous interpretation (m - interpretation for short) which does not allow the existence of pseudoterminals. As a consequence of the modification of the interpretation mechanism we are able to give necessary and sufficient conditions for the m - completeness of short and simple EPOL forms. Finally the fact that many fundamental results also hold under m - interpretation shows the close relation to the ordinary interpretation mechanism whereas complications in carrying over the technique of isolation to m - interpretation gives a good feeling of what really happens when avoiding pseudoterminals.

We will denote an EOL system G by $G = (V,\Sigma,P,S)$ where V is the total, Σ the terminal alphabet, P the set of productions and S the startsymbol. G is called <u>short</u> if $\alpha \to x \in P$ implies $|x| \leq 2$, <u>simple</u> if $\mathrm{card}(V - \Sigma) = \mathrm{card}(\Sigma) = 1$, <u>synchronized</u> if $a \overset{+}{\Longrightarrow} x$ implies $x \notin \Sigma^*$ for every $a \in \Sigma$ and <u>looping</u> if $\alpha \overset{+}{\Longrightarrow} \alpha$ holds for some $\alpha \in V$.

For a precise definition of the notions used we refer to [H], [RS] and [MSW1]. However, we would like to give the fundamental definition of L form theory:

An EOL form F is an EOL system, $F = (V,\Sigma,P,S)$. An EOL system $F' = (V', \Sigma',P',S')$ is called an interpretation of F (modulo μ), if μ is a finite substitution defined on V and (i) to (v) hold:

(i) $\mu(A) \subseteq V' - \Sigma'$ for each $A \in V - \Sigma$;

(ii) $\mu(a) \subseteq \Sigma'$ for each $a \in \Sigma$;

(iii) $\mu(\alpha) \cap \mu(\beta) = \phi$ for all $\alpha \neq \beta$ in V;

(iv) $P' \subseteq \bigcup_{\alpha \to x \in P} \{\beta \to y \mid \beta \in \mu(\alpha),\ y \in \mu(x)\}$;

(v) $S' \in \mu(S)$.

In this case we write $F' \lhd F(\mu)$. $\mathcal{L}(F) := \{L(F') \mid F' \lhd F\}$ is the family of languages generated by F.

Examples of EOL systems are specified by listing the productions where small letters are used to denote terminals, capital letters to denote nonterminals and S to indicate the startsymbol. Many results in this paper are only sketched. A detailed version is available as [Ai] and has been submitted for publication elsewhere.

Underlying Philosophy

Rewriting systems have originally been introduced by Thue. Thue did not make a distinction between a terminal and a nonterminal alphabet as it has become customary in formal language theory. This distinction is due to three reasons: firstly, the introduction of nonterminals has a linguistic motivation since nonterminals can be viewed as representations of syntactic classes. Secondly, the variety of languages obtained by grammars is essentially increased with the use of nonterminals, cf. [MSW3] where pure grammars (that are grammars without nonterminal symbols) are investigated. Thirdly, nonterminals are necessary to obtain strong closure properties, e.g. one can show that the class of pure CF languages is an anti - AFL, whereas it is well - known that the class of CF languages (which differs from the first one only in the existence of nonterminal symbols) is a full AFL.

When considering parallel rewriting we may observe similar conditions: originally parallel rewriting has been introduced in [L] to describe the development of cell growth in simple organisms. These rewriting systems, so - called L systems, do not use nonterminals. However, it turned out that the introduction of nonterminal symbols in [He] involves similar advantages concerning the increase of languages obtainable and closure properties as mentioned above in the case of grammars, cf. [S] and [He]. The extension of OL systems to EOL systems by introducing nonterminals was found mathematically tractable and interesting. Moreover, we can justify the notion of extended OL systems from a biological point of view pointed out in [HR]: the family of languages of recurrence systems, which is of biological interest, equals the family of EOL languages. A final argument for considering EOL systems is the equivalence of the class of EOL languages and the class of codings of OL languages, cf. [ER], and the significance of codings for biological observations.

EOL systems differ from CF grammars in two ways: parallel rewriting is used rather than sequential rewriting and in EOL systems there exist productions also for terminal symbols. One could suspect that the latter is a natural consequence of the constraint of parallel rewriting: if there are no productions for terminal symbols, any derivation in an EOL system will stop whenever a terminal symbol is generated. Note that this situation is simulated exactly by synchronized EOL systems. Thus, as far as the generated languages are concerned, the existence of terminal productions is quite insignificant since it is well - known that for any EOL language L there exists a synchronized EOL system F such that L(F) = L, cf. [HR]. However, when working with EOL families, it turns out that terminal productions indeed lead to additional language families as shown in [MSW1].

When introducing nonterminal symbols for CF grammars, the character of all symbols not contained in the set of nonterminals is really "terminal" in the sense that each of these symbols actually occurs in some word of the generated language, provided the grammar is reduced. Clearly, this is due to sequential rewriting in CF grammars. The situation becomes more complicated in the case of EOL systems as demonstrated by the following example: let F be defined by the productions $S \to aS$, $S \to b$, $a \to b$ and $b \to b$. Clearly, $L(F) = b^+$. Although the symbol a is explicitely specified as member of the terminal alphabet, it does not occur in any word of the language which is caused by the parallel mode of rewriting.

<u>Definition:</u> Let $F = (V,\Sigma,P,S)$ be an EOL system. A symbol $a \in \Sigma$ is called a pseudoterminal iff $a \notin alph(L(F))$. PS(F) denotes the set of pseudoterminals of F.

The existence of pseudoterminals has been observed in a number of proofs in the past, cf. [MSW2] and [AiM]. In these cases pseudoterminals often lead to rather nasty complications. However, pseudoterminals play an important role for some results concerning the completeness of

EOL forms settled in [CM] and [CMO]. Also the quite surprising and somewhat pathological result of Theorem 3.4 in [AiM] seems to depend essentially on the existence of pseudoterminals. The aim of this paper, namely to consider EOL forms with restricted occurrence of pseudoterminals, is due to two reasons: the first one is to avoid complications as mentioned above and is a rather pragmatic one. The second reason becomes obvious when analysing the proofs of Theorem 2.4 and Theorem 2.5 in [CM]. These theorems establish the existence of complete EOL forms which do not contain a nonterminal chain - production, i.e. a production of the type $A \to B$ where A and B are nonterminals. This result is shown by a construction which uses pseudoterminals, i.e. terminal symbols with nonterminal character, to generate necessary nonterminal chains, thus veiling and falsifying in a certain way our knowldge about the structure of derivation trees which are necessary to generate all EOL languages. Indeed, Theorem 1 shows that such nonterminal chain - productions are necessary for completeness when suppressing pseudoterminals. We think that the mentioned results in [AiM], [CM] and [CMO] are not due to the structure of EOL systems in the first place but due to a weakness in the definition of EOL systems.

<u>Definition:</u> An EOL system $F = (V,\Sigma,P,S)$ is called <u>marvellous</u> if Σ contains no pseudoterminals.

The following lemma, which is easy to prove, shows that the generative capacity of EOL systems is not affected by this definition.

<u>Lemma 1:</u> For every EOL language L there exists a marvellous EOL system F such that $L(F) = L$. □

When dealing with EOL forms one easily checks that the form being marvellous is not sufficient to assume that all interpretations are marvellous. A general relation beween the sets of pseudoterminals of the form and its interpretations, respectively, is established by the following lemma.

<u>Lemma 2:</u> Let $F = (V,\Sigma,P,S)$ be an EOL form. For every interpretation $F' = (V',\Sigma',P',S') \triangleleft F(\mu)$ there holds: $\mu(PS(F)) \subseteq PS(F')$. □

We next present two examples. The first one shows that the inclusion of Lemma 2 may be proper; the second on shows that despite Lemma 2 interpretations of forms containing pseudoterminals may be marvellous (due to the fact that $\mu(PS(F)) = \phi$ may hold).

<u>Example 1:</u>

F: $S \to aS \mid a \mid b;\ a \to b;\ b \to b.$

F': $S \to aS \mid b;\quad a \to b;\ b \to b.$

Clearly, $F' \lhd F(\mu)$, $PS(F) = \phi$ and $PS(F') = \{a\}$.

Example 2:

F: $S \to a \mid bS;\ a \to a;\ b \to S$.

F': $S \to a;\ a \to a$.

Again, $F' \lhd F(\mu)$, and $PS(F) = \{b\}$, $PS(F') = \phi$.

Note that the generation of pseudoterminals via interpretation is crucial to the proof of the normal form result for EOL systems in [CM]. For example, the complete form G specified by the productions $S \to a \mid aS \mid Sa$, $a \to a \mid S \mid SS$ clearly does not contain pseudoterminals. However, the construction used in the proof of Theorem 2.5 in [CM] uses pseudoterminals which are interpretations of the terminal symbol a. By Lemma 2 and the above example it becomes obvious that the definition of marvellous systems does not suffice for the consideration of EOL forms. Indeed, we also have to modify the mechanism of interpretation, thus getting what we call marvellous or m - interpretation. Before presenting our definition we want to briefly discuss an alternative and why we feel that this alternative is not suitable: the idea of the modification is to allow interpretations of terminals to be nonterminals in the case that the interpreted terminal would have been a pseudoterminal. Thus, we could call an EOL system $F' = (V',\Sigma',P',S')$ a marvellous interpretation of the EOL system $F = (V,\Sigma,P,S)$ modulo μ (in symbols: $F' \lhd_m F(\mu)$) if μ is defined as usual except point (ii) which is altered to:

(ii) for all $a \in \Sigma$ and all $\alpha \in \mu(a)$

$$\alpha \in \begin{cases} V' - \Sigma' & \text{if for all } x' \in SF(F')\ \alpha \in alph(x') \text{ implies } alph(x') \\ & \quad \cap\ \mu(V - \Sigma) \neq \phi \\ \Sigma' & \text{otherwise.} \end{cases}$$

Clearly, the definition guarantees that every interpretation is marvellous. The main drawback of this kind of definition is that it blures the relation between the form and its interpretations. This fact greatly decreases the possibility of using complete forms as normal form results which, however, is one of the main objects in considering completeness of EOL forms. By Lemma 1 one easily checks that for each EOL form F there holds $\mathcal{L}(F) = \mathcal{L}_m(F)$. Thus, even under marvellous interpretation as defined above the form G with productions listed above remains complete. But, although G does not contain nonterminal chain - productions, that result does not imply that every EOL language can be generated by a marvellous EOL system containing no nonterminal chain - production as shown by Theorem 1. Indeed, this type of definition suppresses pseudoterminals in a merely formal way. The character of pseudoterminal symbols is not taken into consideration and thus the main complications which lead to the modification of the interpretation mechanism do not disappear. Let us now define m - interpretation:

Definition: Let $F = (V,\Sigma,P,S)$ and $F' = (V',\Sigma',P',S')$ be marvellous EOL systems. Then F' is called a marvellous interpretation (m - interpretation for short) of F (modulo μ), symbolically $F' \underset{m}{\lhd} F(\mu)$, iff $F' \lhd F(\mu)$. Additionally, $\mathcal{L}_m(F)$ and m - completeness are defined as usual but with respect to m - interpretation.

Remarks: Note that it also has been customary in the past to put constraints on the involved systems when defining interpretations for EOL forms. Since an EOL system must have a complete set of productions, i.e. there has to exist at least one production for each symbol, it follows that not each rewriting system F' obtained from an EOL form by a substitution μ is an EOL system again. In this case we do not have $F' \lhd F(\mu)$ even if μ satisfies conditions (i) to (v) since an interpretation is defined only for EOL systems. In our case additionally to the necessity of considering rewriting systems with a complete set of productions, e.g. EOL systems, we have to take care that F and F' are marvellous. Since it is decidable for every EOL system wether it is marvellous as will be shown in Lemma 3 our definition of m - interpretation is meaningfull and the relation $\underset{m}{\lhd}$ remains decidable. Note further that our definition exactly avoids the introduction of additional pseudoterminals via interpretation. For example, let $\mathcal{M}(F) = \{L(F') \mid F' \lhd F(\mu) \text{ and } \mu(PS(F)) = PS(F')\}$. Clearly, every language in $\mathcal{M}(F)$ can be generated by an interpretation (of F) which does not introduce additional pseudoterminals. Using the technique of Lemma 1 it can be shown that for every EOL form F there exists a marvellous EOL form F_1 such that $\mathcal{M}(F) = \mathcal{M}(F_1)$. This and the result of Lemma 1 show that our solution, which is somewhat more elegant, suffices since neither the generative power of EOL systems nor that of EOL forms (via interpretation) is decreased by considering marvellous forms only.

We want to mention that clearly $\mathcal{L}_m(F) \subseteq \mathcal{L}(F)$ holds for every EOL form F and that the inclusion may be proper. An example for the latter is the form F specified in Example 1. Finally we give the following decidability - result which is easily proved:

Lemma 3: Let $F = (V,\Sigma,P,S)$ be an EOL system. It is decidable for every $a \in V$ wether it is a pseudoterminal. □

Results

Lemma 4: Let $F = (V,\Sigma,P,S)$ be a marvellous EPOL system such that $L(F) = \{a^n b^n a^n b^n \mid n \geq 1\}$. Then $P \cap (V - \Sigma) \times (V - \Sigma) \neq \phi$.

Sketch of proof: We show that the following assumptions lead to a contradiction:

(1) F is a marvellous EPOL system and $L(F) = \{a^n b^n a^n b^n \mid n \geq 1\}$;

(2) $P \cap (V - \Sigma) \times (V - \Sigma) = \phi$.

It is well - known that every EPOL system generating L(F) must be looping. By condition (2) we can show that looping symbols in F must be terminal symbols, i.e. are elements of {a,b} since F is marvellous. Let us choose a to be looping, then at the same time b being looping implies $L(F) \in \mathcal{L}(CF)$. This is a contradiction and thus a is the only looping symbol in F. Intuitively it is clear that the fact that every loop of F has to use the terminal symbol a is too restrictive to allow the generation of a language like L(F). In particular, we show that in any F - derivation tree for a word $a^n b^n a^n b^n \in L(F)$ there occurs a path leading from the root labelled S to a leaf labelled b and containing no node with label a. Since a is the only looping symbol in F the above fact bounds the length of successfull derivations in F which leads to the final contradiction as in [CMO]. □

Lemma 5: Let $F = (V,\Sigma,P,S)$ be a short and marvellous EPOL system such that $L(F) = \{a^{5^n} \mid n \geq 1\}$. Then $P \cap (V - \Sigma) \times (V - \Sigma)^2 \neq \phi$.

Proof: We assume the contrary, i.e. P contains no production of the type $A \to BC$, $\{A,B,C\} \subseteq V - \Sigma$. Note that this implies that every length - increasing production in F involves the terminal symbol a since F is marvellous and thus $\Sigma = \{a\}$. Consider $x = a^5 \in L$. By the above observation and since F is short we have $S \overset{+}{\underset{F}{\Longrightarrow}} x_1 a x_2 \overset{+}{\underset{F}{\Longrightarrow}} a^5$ where $x_1 x_2 \in V^+$. The subderivation $a \overset{+}{\underset{F}{\Longrightarrow}} a$ is impossible since it would imply $L(F) \in \mathcal{L}(CF)$ and we have left $a \overset{+}{\underset{F}{\Longrightarrow}} a^j$, $2 \leq j \leq 4$ since $x_1 x_2 \neq \varepsilon$ and F is propagating. This immediately implies a contradiction since $a^{5j} \notin L(F)$ for $2 \leq j \leq 4$. □

Theorem 1: A simple and short EPOL form $F = (\{S,a\},\{a\},P,S)$ is m - complete iff P contains all of the productions $S \to a$, $S \to S$ and $S \to SS$ and at least one of the productions $a \to S$, $a \to aS$, $a \to Sa$ and $a \to SS$.

Proof: By Example 5.1 in [MSW1] the EPOL form G with productions $S \to a$, $S \to S$, $S \to SS$ and $a \to S$ is complete. When analysing the proof one easily checks that productions for terminal symbols are only used to block the derivation after having generated a terminal symbol in the interpretations of G. By Lemma 1 this implies that G is m - complete, too. It is clear that we may use interpretations of the productions $a \to aS$, $a \to Sa$ or $a \to SS$ also only for blocking if the form contains $S \to a$, $S \to S$ and $S \to SS$. Thus, we may assume that those productions do not cause pseudoterminals in the interpretations of the form. By the above observations, Proposition 1.2 in [CMO] and the Lemmas 4 and 5 the theorem follows immediately. □

Consequences

For a given EOL form $F = (V,\Sigma,P,S)$ let $N = \{x_i \in V^* \mid 1 \leq i \leq n\}$ be a finite set of words such that F contains the derivation $\alpha \overset{+}{\underset{F}{\Rightarrow}} x$ for a fixed symbol $\alpha \in V$ and every $x \in N$. Then it is possible for every $M \subseteq N$ to construct an interpretation F' such that whenever a derivation (α) starts with α and ends with a word over V then (α) contains a word $y \in M$, i.e. the derivations $\alpha \overset{+}{\underset{F}{\Rightarrow}} y$, $y \in M$ have been "isolated". The idea is to rename all symbols occurring in the intermediate steps of the derivations such that the new symbols differ from each other and all of the new symbols differ from the symbols of the form F, cf. the Isolation Lemma in [W]. This renaming is easily done in the case of ordinary interpretation when viewing the renamed symbols as interpretations of the original ones: it does not matter wether the original symbol is a terminal or a nonterminal. Obviously, this changes in the case of m - interpretation. Whenever an intermediate word of the derivation which we want to isolate contains both, terminal and nonterminal symbols, the renaming required in general leads to the introduction of pseudoterminals in the interpretation. Moreover, even the possible context of the intermediate words according to F must be taken into consideration. The basic difficulties which occur when isolating via m - interpretation are the following ones:

(i) Introduction of pseudoterminals caused by renaming.

This may happen inside the isolated derivation if an intermediate word of the derivation which contains a terminal symbol occurs together with a nonterminal symbol in any word generated by the form; outside the isolated derivation pseudoterminals may be introduced if a terminal symbol occurs only together with a nonterminal outside the derivation and thus becomes a pseudoterminal in the interpretation since all other occurrences of the symbol have been renamed.

(ii) Introduction of pseudoterminals caused by eliminating productions.

If we isolate a derivation $\alpha \overset{+}{\underset{F}{\Rightarrow}} x$ then clearly any production for