ELSEVIER Theoretical Computer Science 202 (1998) 231–244

Theoretical Computer Science

Note

Comparisons of Parikh's condition to other conditions for context-free languages

G. Ramos-Jiménez, J. López-Muñoz, R. Morales-Bueno*

E.T.S. de Ingeniería Informática, Universidad de Málaga, Dpto. Lenguajes y Ciencias de la Computación, P.O.B. 4114, 29080-Málaga, Spain

Received May 1997; revised July 1997
Communicated by M. Nivat

Abstract

In this paper we first compare Parikh's condition to various pumping conditions – Bar-Hillel's pumping lemma, Ogden's condition and Bader-Moura's condition; secondly, to interchange condition; and finally, to Sokolowski's and Grant"s conditions. In order to carry out these comparisons we present some properties of Parikh's languages. The main result is the orthogonality of the previously mentioned conditions and Parikh's condition. © 1998—Elsevier Science B.V. All rights reserved

Keywords: Context-free languages; Parikh's condition; Pumping lemmas; Interchange condition; Sokolowski's and Grant's condition

1. Introduction

The context-free grammars and the family of languages they describe, context free languages, were initially defined to formalize the grammatical properties of natural languages. Afterwards, their considerable practical importance was noticed, specially for defining programming languages, formalizing the notion of parsing, simplifying the translation of programming languages and in other string-processing applications. It is very useful to discover the internal structure of a formal language class during its study. The determination of structural properties allows us to increase our knowledge about this language class. An additional benefit is obtained when a particular property is found to be easily testable; it then becomes a convenient tool for proving that some languages do not belong to this class. In Fig. 1 we show a classification of the most well-known conditions for context free languages.

* Corresponding author. E-mail: morales@lcc.uma.es.

PII S0304-3975(97)00262-4

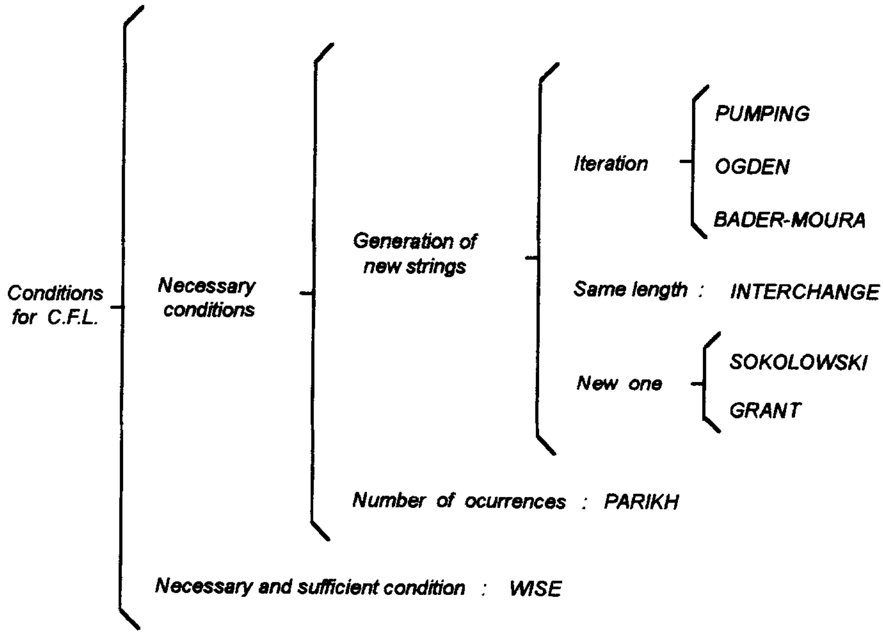

Fig. 1. Classification of conditions for context-free languages.

When there are several conditions in the deepest level, within this classification, each of them is a generalization of the previous condition. So, Bader-Moura's condition is a generalization of Ogden's, and this in turn is a generalization of the pumping condition. In the same way, Grant's condition is a generalization of Sokolowski's.

We observe that most of the lemmas provide only necessary conditions, except Wise's condition [19] but this result includes a construction that is not effective. Therefore, every condition characterizes a family of languages that strictly includes the CFL (context-free languages) class. Thus, it is interesting to compare different conditions in order to show their relationships of inclusion or intersection.

Some of the comparative studies concerning the different conditions are [4, 5, 10, 12]. Among these we recommend [5, 10]. Boonyavatana and Slutzki [5] compare the interchange condition of Ogden, Ross and Winklmann to various pumping conditions: the classic pumping condition of Bar-Hillel, Perles and Shamir; Ogden's condition; generalized Ogden's condition of Bader and Moura; linear versions of the previously mentioned conditions and the Sokolowski-type conditions. Also, they formulated an interchange condition for linear context-free languages and compared it to the other conditions. The same authors [10] carry out a systematic investigation of the relationships between various pumping properties, the interchange condition, and Sokolowski's and the extended Sokolowski's condition of Grant.

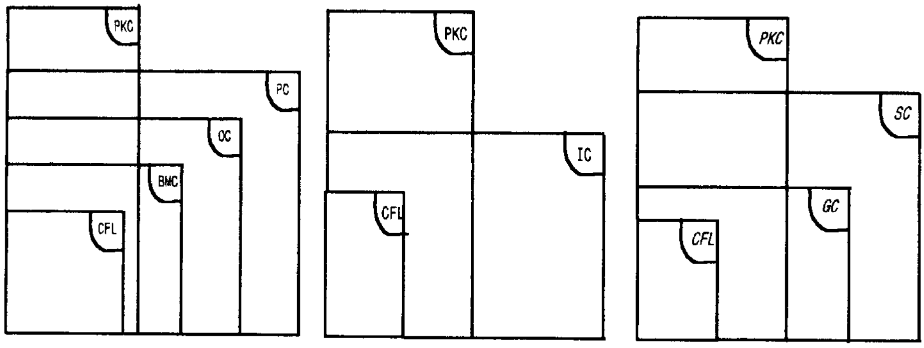

Fig. 2. Comparisons of Parikh's condition.

None of these articles have compared Parikh's condition to the other ones. That is the aim of our paper. We compare Parikh's condition to pumping conditions (Bar-Hillel's, Ogden's and Bader-Moura's), the interchange condition and Sokolowski's and Grants's conditions, and we prove that Parikh's condition is orthogonal to all of them, as shown in Fig. 2. Specifically, we find languages for each of the zones of that figure, where the significance of each zone is described in the subsequent paragraph concerning notation.

The paper is organized as follows: in Section 2 we present the basic definitions and the introductory results. In Section 3, using the outcomes of Section 2, we compare Parikh's condition with the pumping condition. In Section 4, we briefly compare Parikh's condition with the interchange and, Sokolowski's and Grant's condition. Each zone in Fig. 2 is identified by the following notation:

CFL: Context-Free languages	PKC: Parikh's condition
PC: Pumping condition	OC: Ogden's condition
BMC: Bader-Moura's condition	IC: Interchange condition
SC: Sokolowski's condition	GC: Grant's condition

For any condition C, $C = PKC, PC, \ldots, \mathrm{GC}$, and any alphabet Σ,

$$C(\Sigma) = \{L \subseteq \Sigma^*/L \text{ satisfies } C\}.$$

So, as an example, $CFL(\Sigma)$ is the set of context-free languages over Σ.
We omit Σ when there is no ambiguity.

2. Definitions and preliminary results

In this section we present some basic definitions, notations and some preliminary results. We assume that the reader is familiar with the basic theory of context-free languages and so we will only define general concepts and formulate various pumping-type conditions for this language class.

A *context-free grammar* is a construct $G=(N,T,P,S)$ where N and T are two disjoints sets of *nonterminals* and *terminals*, respectively [11]; P is a finite set of *productions* and each production is of the form $A\to\alpha$ where A is a nonterminal and α is a string of symbols from $(N\cup T)^*$; and finally, S is a special nonterminal called the *start symbol* or *axiom*. The *language generated* by G, $L(G)$, is a *context-free language*.

For a word w, $|w|$ denotes its *length*; and ε is the empty word. For a set Q, $\|Q\|$ denotes the *cardinality* of Q. For a language L, L_n is the set of all words of length n in L.

Bar-Hillel, Perles and Shamir (classical) pumping condition. *A language $L\subseteq\Sigma^*$ satisfies PC if there exists a constant n such that if $z\in L$ and $|z|\geqslant n$, then we may write $z=uvwxy$ such that*
(i) $|vx|\geqslant 1$,
(ii) $|vwx|<n$, *and*
(iii) $\forall i\geqslant 0\ uv^iwx^iy\in L$.

A language $L\in PC(\Sigma)$ if L satisfies the pumping condition. We omit Σ when there is no ambiguity.

Ogden's condition. *A language $L\subseteq\Sigma^*$ satisfies OC if there exists a constant n such that if $z\in L$ and we label in it $d(z)$ "distinguished" positions, with $d(z)>n$, then we may write $z=uvwxy$ such that*
(i) $d(u)\times d(v)\times d(w)+d(w)\times d(x)\times d(y)\geqslant 1$,
(ii) $d(vwx)\leqslant n$, *and*
(iii) $\forall i\geqslant 0,\ uv^iwx^iy\in L$.

A language $L\in OC(\Sigma)$ if L satisfies Ogden's condition.

Bader-Moura's condition. *A language $L\subseteq\Sigma^*$ satisfies BMC if there exists a constant n such that if $z\in L$ and we label in it "distinguished" positions $d(z)$ and $e(z)$ "excluded" positions, with $d(z)>n^{e(z)+1}$, then we may write $z=uvwxy$ such that:*
(i) $d(vx)\geqslant 1$ *and* $e(vx)=0$
(ii) $d(vwx)\leqslant n^{e(vwx)+1}$ *and*
(iii) *for every* $i\geqslant 0$, uv^iwx^iy *is in* L.

A language $L\in BMC(\Sigma)$ if L satisfies Bader-Moura's condition.

Pumping lemmas [2, 3, 15]: $CFL(\Sigma)\subset BMC(\Sigma)\subset OC(\Sigma)\subset PC(\Sigma)$.

We now describe the Interchange condition. Put briefly this says that if a language L satisfies it and contains many strings of some fixed length, then parts of these strings may be interchanged, producing new strings which must also be in L. We observe that the pumping conditions predict that increasingly longer strings will be found in the language.

Interchange condition. *A language $L \subseteq \Sigma^*$ satisfies IC if there is a constant c_L such that for any integer $n \geqslant 2$, any subset Q_n of L_n, and any integer m with $n \geqslant m \geqslant 2$ there are $k \geqslant \| Q_n \| /(c_L n^2)$ strings z_i in Q_n with the following properties:*

(i) $z_i = w_i x_i y_i$, $i = 1, \ldots, k$,

(ii) $|w_1| = |w_2| = \cdots = |w_k|$,

(iii) $|y_1| = |y_2| = \cdots = |y_k|$,

(iv) $m \geqslant |x_1| = |x_2| = \cdots = |x_k| > m/2$, *and*

(v) $w_i x_j y_i \in L_n$, $\forall i,j \in \{1, \ldots, k\}$.

A language $L \in IC(\Sigma)$ if L satisfies the IC condition.

Interchange lemma (Ogden, Ross, Winklmann [16]). $CFL(\Sigma) \subset IC(\Sigma)$.

The Sokolowski's criterion says, informally, that if a language L satisfies it and a set of strings A is included in L, then there exists a string that does not belong to A but to L.

Sokolowski's condition. *A language $L \subseteq \Sigma^*$ satisfies SC if for every subset $\Sigma' \subseteq \Sigma$, containing at least two distinct symbols and for u_1, u_2, $u_3 \in \Sigma^*$, if $\{u_1 x u_2 x u_3 \mid x \in \Sigma'^+\} \subseteq L$ then there are two distinct words x', $x'' \in \Sigma'^+$, such that $u_1 x' u_2 x'' u_3 \in L$.*

A language $L \in SC(\Sigma)$ if L satisfies the SC condition.

Sokolowski's lemma (Sokolowski [18]). $CFL(\Sigma) \subset SC(\Sigma)$

This result provides quick and clear proofs that languages like Pascal, Modula-2, etc. are not context-free languages.

Grant observed that in the Sokolowski's proof it is not neccesary to consider strings of the form $u_1 x u_2 x u_3$. Strings $u_1 x_1 u_2 x_2 u_3$ are sufficient with the condition that x_1 and x_2 satisfies some binary relation which is verified for arbitrary long strings.

We need two concepts:

- v' is *previous* to v, $v' < v$, iff v' is obtained from v by omission of at least one letter.
- For $u,v \in \Sigma^*$, $m > 0$, $End(u, v, m)$ (respectively $Beg(u,v,m)$) is true iff v is obtained from u by omitting at least one letter from the last (resp. first) m elements of u.

Grant's condition. A language $L \subseteq \Sigma^*$ satisfies GC if for a binary relation R over Σ^*, satisfying

(i) $\forall m\ \exists x_1 \exists x_2 [\,|x_1|, |x_2| > m \wedge R(x_1, x_2)]$,

(ii) $\{u_1 x_1 u_2 x_2 u_3 \mid R(x_1, x_2)\} \subseteq L$,

then

$$\exists m\ \forall x_1 x_2\ [R(x_1,x_2) \wedge |x_1|, |x_2| > m \rightarrow \exists \alpha_1 \alpha_2 (u_1 \alpha_1 u_2 \alpha_2 u_3) \in L$$

$$\wedge ((\alpha_1 < x_1 \wedge \alpha_2 = x_2) \vee (\alpha_2 < x_2 \wedge \alpha_1 = x_1)$$

$$\vee (End\ (x_1, \alpha_1, m) \wedge Beg(x_2, \alpha_2, m)))]$$

A language $L \in GC(\Sigma)$ if L satisfies the GC condition.

Grant's lemma [8]. $CFL(\Sigma) \subset GC(\Sigma) \subset SC(\Sigma)$.

We now consider Parikh's condition. This condition refers to the global structure of the strings of the language L. We consider the number of times that each symbol appears in a string of L. Let us focus on those numbers forming a vector. If L is infinite then we obtain infinite vectors. Parikh's condition claims that such a set of vectors has a simple structure.

For an alphabet Σ with r symbols, $\Sigma = \{a_1, \ldots, a_r\}$, we define $\#_i(w)$, $w \in \Sigma^*$, as the number of times that a_i occurs in w.

We also define $\psi : \Sigma^* \rightarrow \mathbb{N}^r$, called *Parikh's application*, as

$$\psi(w) = (\#_1(w), \#_2(w), \ldots, \#_r(w)).$$

Let L be a language, $L \subseteq \Sigma^*$, we define $\psi(L) = \{\psi(w) : w \in L\}$.

Let the vectors be $V_0, V_1, \ldots, V_k \in \mathbb{N}^r$. The subset $A \subseteq \mathbb{N}^r$ is *linear* if

$$A = \{V_0 + X_1 V_1 + \cdots + X_k V_k : X_i \in \mathbb{N},\ i = 1, \ldots, k\}.$$

One set S is *semilinear* if it is a finite union of linear sets.

Parikh's condition. *A language $L \subseteq \Sigma^*$ satisfies PKC if $\psi(L)$ is semilinear. A language $L \in PKC(\Sigma)$ if L satisfies PKC; i.e.,*

$$PKC(\Sigma) = \{L \subseteq \Sigma^* \mid L \text{ is Parikh}\} = \{L \subseteq \Sigma^* \mid \psi(L) \text{ is semilinear}\}.$$

Trivially, Σ^* and Σ^+ belong to $PKC(\Sigma)$.

Parikh's lemma [17]. $CFL(\Sigma) \subset PKC(\Sigma)$.

Parikh's lemma has a pumping character because for its proof, a pumping process is necessary in the derivation trees; nevertheless, this condition is different from pumping conditions as we will see in Section 3.

It is known [7, 13] that Parikh's languages over Σ are the rational subsets of the free commutative monoid generated by Σ; and so, Parikh's results can be stated in the following form [1, Theorem 2.8]: "Any context-free set in the commutative monoid is rational."

We show now closure results of Parikh's languages. Considering the definition and properties of the rational sets we state the following results [7, 13].

Theorem 1. (a) *The rational sets are closed under concatenation and union;*
(b) *The rational sets are closed under direct morphisms;*
(c) *The intersection of a rational set with a recognizable set is a rational set.*

We show now the converse with respect to concatenation and union over disjoint alphabets.

Theorem 2. *Let $L_1 \subseteq \Sigma_1^*$ and $L_2 \subseteq \Sigma_2^*$ be two languages over disjoint alphabets,* where $\|\Sigma_1\| = r$ *and* $\|\Sigma_2\| = s$:
(a) *if $L_1L_2 \in PKC(\Sigma_1 \cup \Sigma_2)$ then $L_1 \in PKC(\Sigma_1)$ and $L_2 \in PKC(\Sigma_2)$;*
(b) *if $L_1 \cup L_2 \in PKC(\Sigma_1 \cup \Sigma_2)$ then $L_1 \in PKC(\Sigma_1)$ and $L_2 \in PKC(\Sigma_2)$.*

Proof. Let $\pi_{1,r}^{r+s}: \mathbb{N}^{r+s} \to \mathbb{N}^r$ and $\pi_{r+1,r+s}^{r+s}: \mathbb{N}^{r+s} \to \mathbb{N}^s$ be the projections with respect to the first r components and the last s components, respectively. These projections are morphisms and can be extended to sets of vectors. Thus, by Theorem 1(b) we obtain:

(a) $\pi_{1,r}^{r+s}(\psi(L_1L_2)) = \psi(L_1)$ is semilinear, hence $L_1 \in PKC(\Sigma_1)$,
$\pi_{r+1,r+s}^{r+s}(\psi(L_1L_2)) = \psi(L_2)$ is semilinear, hence $L_2 \in PKC(\Sigma_2)$.
(b) $\pi_{1,r}^{r+s}(\psi(L_1 \cup L_2)) = \psi(L_1) \cup \psi(\varepsilon)$ is semilinear, hence $L_1 \in PKC(\Sigma_1)$,
$\pi_{r+1,r+s}^{r+s}(\psi(L_1 \cup L_2)) = \psi(L_2) \cup \psi(\varepsilon)$ is semilinear, hence $L_2 \in PKC(\Sigma_2)$. □

Definition 1 (Hewett and Slutzki [10]). Let Σ be an alphabet. Let f, g be two symbols, $f, g \notin \Sigma$. Let $\mathbb{P}(\Sigma^*)$ be the class of languages over Σ. We define four operations from $\mathbb{P}(\Sigma^*)$ to $\mathbb{P}(\Sigma^*)$ as follows: For each $L \in \mathbb{P}(\Sigma^*)$

$$a(L) = L\{f^n g^n \mid n \geqslant 1\} \cup \Sigma^*\{f^n g^m \mid n, m \geqslant 1,\ n \neq m\} \cup \Sigma^*,$$

$$r(L) = L\{f^n g^n \mid n \geqslant 1\} \cup \Sigma^*\{f^n g^m \mid n \neq m\},$$

$$e(L) = L\{f^n g^n \mid n \geqslant 1\} \cup \Sigma^*,$$

$$s(L) = \{f^n z g^n \mid z \in L,\ n \geqslant 1\} \cup \Sigma^*.$$

Notation. In the following pages we will represent $x(L)$ as L^x, for $x = a, r, e, s$.

Lemma 1. *If $L \in PKC(\Sigma)$ then $L\{f^n g^n \mid n \geqslant 1\}$, $\Sigma^*\{f^n g^m \mid n \neq m;\ n, m \geqslant 1\}$ and $\Sigma^*\{f^n g^m \mid n \neq m\} \in PKC(\Sigma)$.*

Proof. The sets $\{f^n g^n \mid n \geqslant 1\}$, $\{f^n g^m \mid n \neq m;\ n, m \geqslant 1\}$ and $\{f^n g^m \mid n \neq m\}$ are context-free languages, hence they verify PKC. The PKC class is closed under concatenation (Theorem 1). □

Theorem 3. *If* $L \in PKC(\Sigma)$ *then* L^a, $L^r \in PKC(\Sigma \cup \{f, g\})$.

Proof. From Lemma 1 and Theorem 1. □

The following result is stronger than previous ones because it provides a necessary and sufficient condition relating L and L^e.

Theorem 4. $L \in PKC(\Sigma)$ *if and only if* $L^e \in PKC(\Sigma \cup \{f, g\})$.

Proof. The "only if" is from Theorem 1(a) and Lemma 1.

(If) We suppose that $L^e \in PKC(\Sigma \cup \{f, g\})$.
$L^e = L_1 \cup L_2$, where $L_1 = L\{f^n g^n \mid n \geqslant 1\}$ and $L_2 = \Sigma^*$.
Let $\psi : \Sigma^* \cup \{f, g\} \to \mathbb{N}^{r+2}$, where $\|\Sigma\| = r$.
$\psi(L^e) = X$ is semilinear (by hypothesis) and $\psi(L_2) = \psi(\Sigma^*) = Y$, which is obviously semilinear. Thus, by Theorem 5.8.2 [6], $X - Y$ is semilinear.
$X - Y = \psi(L_1)$, hence $L_1 \in PKC(\Sigma \cup \{f, g\})$ and, by Theorem 2, $L \in PKC(\Sigma)$. □

Finally, we study the s-operation.

Lemma 2. $\psi(L^e) = \psi(L^s)$.

Proof. Each word belonging to L^s is, obviously, simply a permutation of one word belonging to L^e, and viceversa. □

Theorem 5. $L \in PKC(\Sigma)$ *if and only if* $L^s \in PKC(\Sigma \cup \{f, g\})$.

Proof. From Theorem 4 and Lemma 2. □

3. Comparison of Parikh's condition to pumping conditions

In this section we will compare Parikh's condition to pumping conditions. The final results are depicted in Fig. 3. In this figure, each rectangle represents the set of languages that satisfy the corresponding condition. We show that none of the zone A1, B1, C1, D1, E1, F1, G1 and H1 are empty.

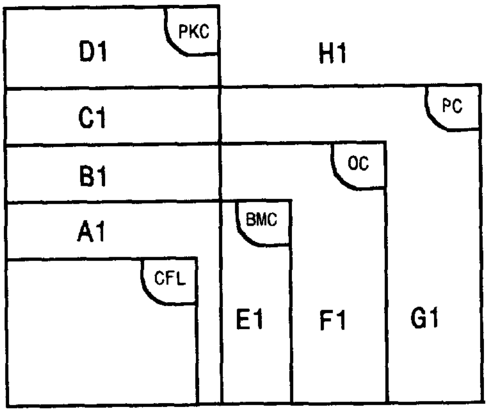

Fig. 3. Parikh's condition compared to pumping conditions.

Proposition 1. *$L1 \in PKC \cap (BMC\text{-}CFL)$; i.e., to zone $A1$, where $L1$ is defined as follows* [2]: $L1 = \{z \in \{a,b\}^* \mid (\exists q : z = (ab)^q) \Rightarrow (q \text{ prime})\}$.

Proof. $L1 \in (BMC\text{-}CFL)$ [2].

We show now that $L1 \in PKC$: we notice that, e.g., words of the form $(ab)^n \notin L1$ if n is not a prime number, but words of the form $a^n b^n \in L1$. So, $\psi(L_1) = \mathbb{N}^2 - \{(0,0)\}$ is semilinear with only two linear sets. □

Proposition 2. *$L2 \in PKC\text{-}PC$; i.e., to zone $D1$, where $L2$ is defined as follows* [10]:

$$L2 = \{a^p b^p c^r d^r \mid 1 \leqslant p \leqslant r\} \cup \{a^p b^q c^r d^s \mid 1 \leqslant q < p \text{ and } p - q \leqslant \max(r,s)\}$$

$$\cup \{a^p b^q c^r d^s \mid 1 \leqslant r < s \text{ and } s - r \leqslant \max(p,q)\} \cup \{a^p b^q c^{s+1} d^s \mid p,q,s \geqslant 1\}.$$

Proof. $L2 \notin PC$ [10]. It is an easy exercise to verify that $L2 \in PKC$. □

Proposition 3. *$L3 \in PKC \cap (OC\text{-}BMC)$; i.e., to zone $B1$, where $L3$ is defined as follows* [10]: $L3 = L2^r$.

Proof. $L3 \in (OC\text{-}BMC)$ [10]. $L3 \in PKC$, from Proposition 2 and Theorem 3. □

Proposition 4. *$L4 \in PKC \cap (PC\text{-}OC)$; i.e., to zone $C1$, where $L4$ is defined as follows* [10]: $L4 = L2^a$.

Proof. $L4 \in (PC\text{-}OC)$ [10]. $L4 \in PKC$, from Theorem 3. □

Proposition 5. *$L5 \in \overline{(PKC \cup PC)}$; i.e., to zone $H1$, where $L5 = \{a^p \mid p$ prime$\}$.*

Proof. Regular, context-free and Parikh's languages define the same class if we consider alphabets with only one letter [9, 14].

$L5$ does not verify the pumping lemma for regular languages [14]. Then $L5$ is not a regular language; therefore $L5 \notin PC$ and $L5 \notin PKC$. □

Proposition 6. *$L6 \in (PC\text{-}OC)\text{-}PKC$; i.e., to zone $G1$, where $L6$ is defined as follows* [5]: $L6 = L5^e$.

Proof. $L6 \in (PC\text{-}OC)$ [5]. $L6 \notin PKC$, from Theorem 4 and Proposition 5. □

Proposition 7. *$L8 \in (OC\text{-}BMC)\text{-}PKC$; i.e., to zone $F1$, where $L8$ is defined as follows* [10]: $L8 = L7^r$; $L7 = \{a^k \mid k \neq n!, \forall n \geqslant 1\}$.

Proof. $L8 \in (OC\text{-}BMC)$ [10].

We show that $L8 \notin PKC$: $L8 = L7\{f^n g^n \mid n \geqslant 1\} \cup \Sigma^*\{f^n g^m \mid n \neq m\}$.

Let us suppose that ψ $(L8)$ is semilinear; i.e., rational.

$$\psi(L8) = \{(x, n, m) \mid n \neq m\} \cup \{(x, n, m) \mid x \neq k!\}.$$

Let us K be the set $K = \{(z, 1, 1)\}$. K is a reconizable set (it corresponds to the regular language a^*fg). Then $\psi(L8) \cap K = \{(z, 1, 1) \mid z \neq k!\}$ is rational (Theorem 1).

By projecting with respect to the first component, we obtain that $S = \{z \mid z \neq k!\}$ is rational. S is a subset of $\mathbb{N}$, then S must be recognizable and $\bar{S} = \{z \mid z = k!\}$ too. But this is absurd because the language $\{a^{n!}\}$ is not regular [9]. □

We study now the zone $E1$. We need some previous results.

Definition 2. Let $a \geqslant 1$ and $b \geqslant 0$ be two integers and let c, f and g tree letters. We define the following language:

$$PRIMES(a, b) = \{f^i c^p g^i \mid i \geqslant 1,\ p \text{ prime and } p \leqslant ai + b\}$$
$$\cup \{f^n c^k g^m \mid n \neq m \text{ and } k \geqslant 0\}.$$

We show now that the language $PRIMES(a, b)$ is in zone $E1$; i.e., $PRIMES(a, b)$ verifies Bader-Moura's condition, but it does not verify the Parikh's condition. The proof is structured in three lemmas. The first one shows that the language $PRIMES(a, b) \in BMC$; the second one is intermediate to show in the third lemma that language does not verify Parikh's condition.

Lemma 3. *PRIMES*$(a, b) \in BMC$.

Proof. Since the second "part" of PRIMES(a, b) is a context-free language, we only need to consider $z = f^i c^p g^i$ where p is a prime, $p \leqslant ai + b$ for some $i \geqslant 1$ and z has a marking such that $d(z) > n^{e(z)+1}$ where $n = \max\{a + 2, b\} + 1$.

If there exist some distinguished non excluded positions among f's, then let v be the leftmost distinguished non excluded position in f^i, let w the symbol that follow v, $x = \varepsilon$ and define u and y accordingly.

So, the three conditions of BMC are verified:

(i) $d(vx) = d(v) = 1$ and $e(vx) = e(v) = 0$

(ii) $d(vwx) \leqslant 2 \leqslant n^{e(vwx)+1}$, because $n \geqslant 4$ and

(iii) for every $i \geqslant 0$, uv^iwx^iy is in L, because for $i \neq 1$ the number of f's is different of the number of g's, and the pumped word belongs to the second "part".

If there exist some distinguished non excluded positions among g's, then let x be the rightmost distinguished non excluded position in g^i, let w be the symbol before x, $v = \varepsilon$ and define u and y accordingly.

Thus, the three conditions of BMC are verified (similar to the above reasoning).

Finally, if there are no distinguished non-excluded positions among f's or g's, then there must exist some non excluded position among f's (otherwise we would have $e(z) \geqslant i$ which implies that $n^{e(z)+1} \geqslant n^{i+1} \geqslant n(i + 1) = ni + n > (a + 2)i + b = ai + b + 2i \geqslant p + 2i = |z| \geqslant d(z)$, contradicting our assumption $d(z) > n^{e(z)+1}$). Thus, let v be the leftmost non excluded position in f^i. Let x be the leftmost distinguished non excluded position in c^p (since if all positions in c^p are distinguished and excluded, and these are the only ones distinguished, our assumption $d(z) > n^{e(z)+1}$ is not verified; the same contradiction is obtained if we consider the positions among f's and g's, that can be distinguished and excluded, or excluded and non distinguished). Finally, we define u, w, y accordingly.

In this way, the three conditions of BMC are verified:

(i) $d(vx) = 1$ (v is non distinguished, and x is distinguished) and $e(vx) = 0$ (v and x are non excluded)

(ii) $d(vwx) \leqslant e(vwx) + 1 < n^{e(vwx)+1}$ and

(iii) for every $i \geqslant 0$, uv^iwx^iy is in L, because for $i \neq 1$ the number of f's is different of the number of g's, and the pumped word belongs to the second "part". □

Definition 3. *Let* $a, b \in \mathbb{N}$, $a \geqslant 1$, $b \geqslant 0$. *We define*

$$P(a,b) = \{(i, p, i) \in \mathbb{N}^3 / i \geqslant 1,\ p \text{ is prime},\ p \leqslant ai + b\}.$$

Lemma 4. $\forall a \geqslant 1$, $\forall b \geqslant 0$, $P(a,b)$ *is not semilinear.*

Proof. Let us suppose that $P(a,b)$ is semilinear. Then, the projection with respect to the second component must be semilinear (Theorem 1). But this projection is

$\{p \in \mathbb{N} \mid p \text{ prime}\}$, that is not semilinear (Proposition 5). Therefore, $P(a,b)$ is not semilinear. □

Lemma 5. *$PRIMES(a, b) \notin PKC$.*

Proof. Let us suppose that $PRIMES(a,b) \in PKC$; then, $X = \psi(PRIMES(a,b))$ is semilinear. Let $Y = \psi(\{f^n c^k g^m \mid n \neq m, k \geqslant 0\})$. We know that Y is semilinear because the language is context-free. By Theorem 5.8.2 [6], $X - Y$ is semilinear. Since the elements of the second part of $PRIMES(a,b)$ do not overlap with those of the first part, we obtain that $X - Y = P(a,b)$ is semilinear, which is a contradiction in respect to the previous lemma. □

Theorem 6. *$PRIMES(a,b) \in BMC\text{-}PKC$; i.e., to zone $E1$.*

Theorem 7. *Zones $A1$, $B1$, $C1$, $D1$, $E1$, $F1$, $G1$ and $H1$, in Fig. 3 are non-empty.*

4. Comparison of Parikh's condition to interchange condition, and Sokolowski's and Grant's condition

In a similar way to Section 3 we do other comparisons. The final results are depicted in Fig. 4.

This section is very brief. The proofs of the following two theorems only includes the relation of languages that are in each zone. The languages used are in the literature. The complete proofs use the closure results of Section 2 and, in order to prove that a language verifies PKC, it is an easy exercise to obtain the suitable semilinear set.

The notation $L_i \dot{\cup} L_j$ (disjoint union) represents: $L_i \subseteq \Sigma_1^*$, $L_j \subseteq \Sigma_2^*$, $\Sigma_1 \cap \Sigma_2 = \emptyset$.

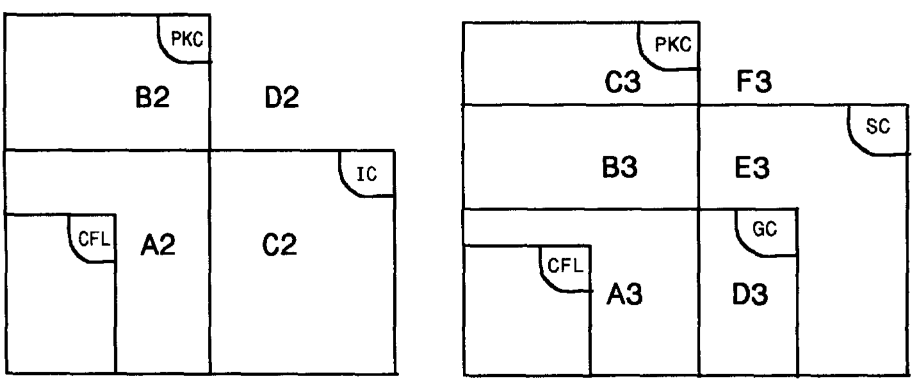

Fig. 4. Parikh's condition compared to Interchange, and Sokolowski's and Grant's condition.

Theorem 8. *Zones A2, B2, C2 and D2 in Fig. 4 are not empty.*

Proof (*sketch*).

$L9 = \{z \in \{a,b\}^* \mid (\exists q\colon z = ab^q) \Rightarrow (q \text{ prime})\}$ [2]; $L9 \in PKC(\{a,b\})$.

$L10 = \{xuu^R \# vv^R y \mid x,u,v,y \in \{a,b,c\}^*\}$ [5]; $L10 \in CFL(\{a,b,c\})$.

$L11 = L9 \dot{\cup} L10$ is in zone $A2$.

$L12 = \{uxxy/x \neq \varepsilon \wedge x,y,u \in \{a,b,c\}^*\}$ [5]; $L12 \in PKC(\{a,b,c\})$.

$L13 = L9 \dot{\cup} L12$ is in zone $B2$.

$L14 = L5 \dot{\cup} L10$ is in zone $C2$.

$L15 = L5 \dot{\cup} L12$ is in zone $D2$. □

Theorem 9. *Zones A3, B3, C3, D3, E3 and F3 in Fig. 4 are not empty.*

Proof (*sketch*).

$L12$ is in zone $A3$.

$L16 = \{a^n b^n c^n \mid n \geqslant 0\}$ is in zone $B3$.

$L17 = \{xx \mid x \in \{a,b,c\}^*\}$ is in zone $C3$.

$L5 = \{a^p \mid p \text{ prime}\}$ is in zone $D3$.

$L18 = L5 \dot{\cup} L16$ is in zone $E3$.

$L19 = L5 \dot{\cup} L17$ is in zone $F3$. □

Acknowledgements

The authors thanks the anonymous referee for his comments and suggestions which improved the presentation as well as the correctness of this paper; specially in Theorem 2, Proposition 7 and Lemma 4.

References

[1] J. Autebert, J. Berstel, L. Boasson, Context-free languages and pushdown automata, in: G. Rozenberg, A. Salomaa (Eds.), Handbook of Formal Languages, Springer, Berlin, 1997.

[2] C. Bader, A. Moura, A generalization of Ogden's lemma, J. ACM 29 (1982) 404–407.

[3] Y. Bar-Hillel, M. Perles, E. Shamir, On formal properties of simple phrase-structure grammars, Zeitschrift für Phonetik, Sprachwissenschaft, und Kommunikationsforschung, 14 (1981) 143–177.

[4] R. Boonyavatana, G. Slutzki, A generalized Ogden's lemma for linear context-free languages, EATCS Bull. 28 (1985) 20–28.
[5] R. Boonyavatana, G. Slutzki, The interchange or pump (di)lemmas for context-free languages, Theoret. Comput. Sci. 58 (1988) 321–338.
[6] S. Ginsburg, The Mathematical Theory of Context Free Languages, McGraw-Hill, New York, 1988.
[7] J.S. Golan, The Theory of Semirings with Applications in Mathematics and Theoretical Computer Science. Pitman Monographs and Surveys in Pure and Applied Mathematics, vol. 54, Longman Sci. Tech., 1992.
[8] P.W. Grant, Extensions of Sokolowski's theorem to prove languages are not context-free or not regular, Internat. J. Comput. Math. 11 (1982) 187–198.
[9] M.A. Harrison, Introduction to Formal Languages Theory, Adisson-Wesley, Reading, MA, 1978.
[10] R. Hewett, G. Slutzki, Comparison between some pumping conditions for context-free languages, Math. Systems Theory 21 (1989) 223–233.
[11] J.E. Hopcroft, J.D. Ullman, Introduction to automata theory, languages, and Computation, Addison-Wesley, Reading MA, 1979.
[12] S. Horváth, A comparison of iteration conditions on formal languages, Proc. Coll. Algebra, Combinatorics and Logic in Computer Science, 1983, pp. 453-483.
[13] W. Kuich, A. Salomaa, Semirings, Automata and Languages, Springer, Berlin, 1988.
[14] H.R. Lewis, C.H. Papadimitriou, Elements of the Theory of Computation, Prentice-Hall, Englewood Cliffs, NJ, 1981.
[15] W. Ogden, A helpful result for proving inherent ambiguity, Math. Systems Theory, 2(3) (1988) 191–194.
[16] W. Ogden, R.J. Ross, K. Winklmann, An "interchange lemma" for context-free languages, SIAM J. Comput., 14(2) (1985) 410–415.
[17] R.J. Parikh, On context-free languages, J. ACM 13(4) (1988) 570–581.
[18] Sokolowski, S., A method for proving programming languages non-context-free, Inform. Process. Lett. 7(3) (1978) 151–153.
[19] D.S. Wise, A strong pumping lemma for context-free languages, Theoret. Comput. Sci. 3 (1978) 359–389.