INFORMATIQUE THÉORIQUE ET APPLICATIONS

LILA KARI
ALEXANDRU MATEESCU
GHEORGHE PAUN
ARTO SALOMAA

On parallel deletions applied to a word

Informatique théorique et applications, tome 29, n° 2 (1995), p. 129-144

<http://www.numdam.org/item?id=ITA_1995__29_2_129_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 29, n° 2, 1995, p. 129 à 144)

ON PARALLEL DELETIONS APPLIED TO A WORD (*)

by Lila KARI ([1]), Alexandru MATEESCU ([1]),
Gheorghe PAUN ([2]) and Arto SALOMAA ([1])

Communicated by J. BERSTEL

Abstract. – *We consider sets arising from a single word by parallel deletion of subwords belonging to a given language. The issues dealt with are rather basic in language theory and combinatorics of words. We prove that every finite set is a parallel deletion set but a strict hierarchy results from k-bounded parallel deletions. We also discuss decidability, the parallel deletion number associated to a word and a certain collapse set of a language, as well as point out some open problems.*

Résumé. – *Nous considérons les ensembles produits à partir d'un mot par l'effacement en parallèle de facteurs appartenant à un langage donné. Ces problèmes sont de nature fondamentale dans la théorie des langages et en combinatoire des mots. Nous prouvons que tout ensemble fini est un ensemble obtenu par effacement parallèle, et qu'il existe une hiérarchie stricte en se bornant à k effacements parallèles. Nous examinons également la décidabilité, le nombre d'effacement parallèle associé à un mot, et un certain ensemble d'écroulement d'un langage. Nous mentionnons aussi quelques problèmes ouverts.*

1. INTRODUCTION

The deletion of specific subwords from a word is an operation basic in language theory.

Left and right derivatives are special cases of this operation. Examples of the wide range of applications of this operation are bottom-up parsing

(*) Received April 1993, revised October 1993.

Research supported by the Academy of Finland, grant 11281, and the Alexander von Humboldt Foundation. All correspondence to Lila Kari.

([1]) Academy of Finland and Department of Mathematics, University of Turku, 20500 Turku, Finland. Present address: Department of Mathematics, University of Western Ontario, London Ontario, NGA SB 7, Canada.

([2]) Institute of Mathematics of the Romanian Academy of Sciences, Str. Academiei 14, 70109 Bucuresti, Romania.

Informatique théorique et Applications/Theoretical Informatics and Applications
0988-3754/95/02/$ 4.00/

(a subword is deleted and replaced by a nonterminal), developmental systems (deletion means the death of a cell or a string of cells) and cryptography (decryption may begin by deleting some "garbage" portions in the cryptotext). A systematic study of various types of deletion operations was begun in [1].

The reader is referred to [3] for unexplained notions in formal language theory. The *empty word* is denoted by λ and the *length* of a word w by $|w|$. Following [1], we define the *deletion* and *parallel deletion* of a language $L \subseteq V^*$ from a word $w \in V^*$ by

$$(\star) \qquad (w \rightarrow L) = \{u_1 u_2 \mid u_1 v u_2 = w,\ v \in L\}$$

$$(\star\star) \qquad (w \Rightarrow L) = \{u_1 u_2 \ldots u_{n+1} \mid n \geq 1,\ u_i \in V^*,\ 1 \leq i \leq n+1,$$
$$w = u_1 v_1 u_2 \ldots u_n v_n u_{n+1},$$
$$\text{for} \quad v_i \in L,\ 1 \leq i \leq n,$$
$$\text{and } u_i \notin V^* (L - \{\lambda\}) V^*,\ 1 \leq i \leq n+1\}.$$

Sets of the forms $(\star)$ and $(\star\star)$ are referred to as *deletion* (D–) *sets*, [2], and *parallel deletion* (PD–) *sets*, respectively. Clearly, sets of the forms $(\star)$ and $(\star\star)$ are always finite.

The operations of deletion and parallel deletion are naturally extended, [1], to the case where w is replaced with a language, but in this paper attention is restricted to $(\star)$ and $(\star\star)$. We investigate problems arising from sets $(\star\star)$ and their modifications, sometimes making comparisons with sets $(\star)$.

2. UNIVERSALITY OF PARALLEL DELETION SETS

Most of the finite sets are not deletion sets. For instance, it is easy to see that neither $\{a, b, c\}$ nor $\{aa, ab, ba, bb\}$ is a deletion set. Characterizations of deletion sets and algorithms for deciding whether or not a given set is a deletion set were given in [2]. It is somewhat unexpected that parallel deletion sets are universal in the sense that every finite language can be viewed as a parallel deletion set.

THEOREM 1: *Every finite language is a parallel deletion set, that is, can be represented in the form* $(\star\star)$.

Proof: If $V = \{a\}$, and $F = \{a^{i_1}, a^{i_2}, \ldots, a^{i_n}\}$, then we denote

$$p = \max \{i_j \mid 1 \leq i \leq n\},$$

and we define

$$w = a^{2p+1},$$
$$L = \{a^{2p+1-i_j} \mid 1 \leq j \leq n\}.$$

As only one string of L can be deleted from w, we obtain $(w \Rightarrow L) = F$.

Consider now V with card $(V) \geq 2$ and take

$$F = \{x_1, x_2, \ldots, x_n\}.$$

We construct

$$w = (x_1\#_1)^2 (x_2\#_2)^2 \ldots (x_{n-1}\#_{n-1})^2 x_n\#_n,$$
$$L = \{(x_j\#_j)^2 \mid 1 \leq j \leq n-1\} \cup \{\#_n\}$$
$$\cup \{\#_j x_j\#_j (x_{j+1}\#_{j+1})^2 (x_{j+2}\#_{j+2})^2$$
$$\ldots (x_{n-1}\#_{n-1})^2 x_n\#_n \mid 1 \leq j \leq n-1\},$$

where $\#_1, \ldots, \#_n$ are new symbols not in V.

From the form of w and of strings in L, it is clear that in every deletion we have to erase either $\#_n$ or a string

$$\#_j x_j\#_j (x_{j+1}\#_{j+1})^2 (x_{j+2}\#_{j+2})^2 \ldots (x_{n-1}\#_{n-1})^2 x_n\#_n,$$

as well as all the remaining substrings $(x_i\#_i)^2$, $1 \leq i \leq j-1$. This implies all symbols $\#_i$, $1 \leq i \leq n$, are erased and only a string x_j remains, $1 \leq j \leq n$. In conclusion, $(w \Rightarrow L) = F$.

Now, take $a, b \in V$, $a \neq b$ (remember that card $(V) \geq 2$) and denote

$$k = \max \{|x_i| \mid 1 \leq i \leq n\}.$$

We replace each occurrence of $\#_i$ in w and in strings of L by $ba^{k+i}b$, $1 \leq i \leq n$. We denote by w', L' the string and the language obtained in this way, respectively. As no string in F can contain a substring a^{k+i}, $1 \leq i \leq n$, the strings $ba^{k+i}b$ behave exactly as the markers $\#_i$, $1 \leq i \leq n$, hence again we have $(w' \Rightarrow L') = F$, which concludes the proof. □

3. A GENERAL UNDECIDABILITY RESULT

Because not every finite set is a deletion set, we face a decision problem that was settled in [2]. An analogous problem does not exist for parallel

deletion sets. However, we can fix the nonempty finite set F in the equation

$$(w \rightarrow L) = F,$$

and ask for an algorithm deciding for a given context-free language L whether or not a solution w exists. If such an algorithm exists, we say that F is *CF-decidable*, otherwise *CF-undecidable*. Similarly, we fix F in the equation

$$(w \Rightarrow L) = F$$

and speak of *CF-p-decidable* ("p" from "parallel") and *CF-p-undecidable* sets F.

It was shown in [2] that $F = \{\lambda\}$ is the only CF-decidable set. Moreover, $\{\lambda\}$ is "CF-universal" in the sense that, for any (nonempty) context-free language L, there is a word w such that $(w \rightarrow L) = \{\lambda\}$. Obviously, the same result holds for parallel deletion as well. In fact, we have

THEOREM 2: *The set $\{\lambda\}$ is CF-p-universal and this is the only CF-p-universal set.*

Proof: Given L context-free, we obtain $(w \Rightarrow L) = \{\lambda\}$ for w one of the shortest strings in L, therefore $\{\lambda\}$ is universal.

Moreover, no set $F \neq \{\lambda\}$ can be CF-p-universal, because for any w we have $(w \Rightarrow V^*) = \{\lambda\} \neq F$. □

In spite of the fact that parallel deletion sets coincide with finite sets, we obtain the same undecidability result as for sequential deletion.

THEOREM 3: *Every finite nonempty set $F \neq \{\lambda\}$ is CF-p-undecidable.*

Proof: Let $F \subseteq V^*$ be a finite language, $F = \{x_1, x_2, \ldots, x_n\}$, with $k = \max \{|x_i| \mid 1 \leq i \leq n\} \geq 1$. If $V = \{a\}$, then we add the symbol b to V (we still denote by V the obtained alphabet), therefore, without loss of generality we may assume card $(V) \geq 2$.

We now proceed as in the proof of Theorem 1 when dealing with alphabets V with card $(V) \geq 2$, namely we construct the string w' and the language L' such that $(w' \Rightarrow L') = F$.

Take now an arbitrary context-free language $L_0 \subseteq V^+$ and consider two new symbols c, d, not in V. We construct the context-free language

$$M = L'' \cup \{c\}\, L_0\, \{c\},$$

where L'' is obtained from L' by substituting the rightmost string $ba^{k+n}b$ corresponding to the marker $\#_n$ in the construction of Theorem 1, by $\{c\}\ V^*\ \{cd\}$. More exactly, $L'' = \sigma\ (L)$ where σ is the substitution defined by:

$$\sigma\ (\#_i) = ba^{k+i}b, \quad 1 \leq i \leq n-1,$$
$$\sigma\ (\#_n) = \{c\}\ V^*\ \{cd\}, \quad \sigma\ (\alpha) = \alpha \text{ otherwise.}$$

Then there exists a string w such that $(w \Rightarrow M) = F$ if and only if $L_0 \neq V^*$ (which is not decidable for arbitrary context-free languages).

Indeed, if $V^* - L_0 \neq \emptyset$, then take $z \in V^* - L_0$ and consider the string

$$w = (x_1 ba^{k+1}b)^2 \ldots (x_{n-1} ba^{k+n-1}b)^2\ x_n czcd.$$

Now, the role of the rightmost marker $\#_n$ is played by $czcd$. As no string of $\{c\}\ L_0\ \{c\}$ appears as a substring of w, in view of the proof of Theorem 1, we obtain $(w \Rightarrow M) = F$.

Assume now that $L_0 = V^*$ and suppose that there is a string w such that $(w \Rightarrow M) = F$.

We distinguish more cases:

(i) w contains at least one occurrence of d. Note that all occurrences of d from w have to be deleted, as otherwise we obtain in $(w \Rightarrow M)$ words which do not belong to F. As d can be deleted only by words from L'', we deduce that the subwords of w containing d have to be of the form $ycvcd$, $y, v \in V^*$. But, in this case, we can also erase from w the word cvc, which leads us to a word in $(w \Rightarrow M)$ still containing a letter d– a contradiction with the form of the strings in F.

(ii) w contains no occurrence of d but contains occurrences of c. Then we can delete from w only strings of $\{c\}\ L_0\ \{c\}$ and strings in L'' containing no occurrence of c (the strings in L'' containing c contain d, too). If w contains an odd number of occurrences of c, then the strings in $(w \Rightarrow M)$ contain an odd number of occurrences of c, contradicting the form of strings in F. If w contains at least 4 occurrences of c, $w = u_1 cu_2 cu_3 cu_4 cu_5$, u_1, u_2, u_3, $u_4 \in V^*$, $u_5 \in (\{c\} \cup V)^*$, then we can remove cu_3c as belonging to $\{c\}\ L_0\ \{c\}$, and irrespective of other deletions, the first occurrence of c in w remains. Hence we obtain a string not in F.

If $w = u_1 cu_2 cu_3$, $u_1, u_2, u_3 \in V^*$, then in order to obtain strings in F we have to remove cu_2c (and this can be done). This implies w is of the form

$$w = y_0 (x_{i_1} ba^{k+i_1} b)^2 y_1 (x_{i_2} ba^{k+i_2} b)^2 y_2 \ldots (x_{i_j} ba^{k+i_j} b)^2 y_j cu_2 c$$
$$y_{j+1} (x_{i_{j+2}} ba^{k+i_{j+2}} b)^2 \ldots y_s (x_{i_{s+1}} ba^{k+i_{s+1}} b)^2 y_{s+1}$$

with $1 \leq i_t \leq n$, $1 \leq t \leq s$, and $y_0 y_1 \ldots y_{s+1} \in F$.

However the strings $ba^{k+i_t}b$ precisely identify the strings in L'' used in such deletions of substrings in w (in $y_0 y_1 y_2 \ldots y_{s+1}$ we cannot have substrings a^{k+i}, $i \geq 1$) hence only one deletion is possible, that is $(w \Rightarrow M)$ contains only one string. The case $F = \{x\}$, $x \neq \lambda$, is handled below.

(iii) w contains no occurrence of c and d. Then, as in the last part of the previous case, we infer that card $(w \Rightarrow M) = 1$.

For the case $F = \{x\}$, $x \neq \lambda$, take again $L_0 \subseteq V^*$ (for V assumed to contain at least two symbols) and construct

$$M = \{c\} V^* \{c\} \cup V^* \{c\} L_0 \{c\} V^*.$$

If $V^* \neq L_0$, then for $z \in V^* - L_0$ we obtain

$$(xczc \Rightarrow M) = \{x\}.$$

If $L_0 = V^*$, then every w with $(w \Rightarrow M) = \{x\}$ must contain an even number of occurrences of c, $w = u_1 cu_2 c \ldots cu_{2t+1}$, $t \geq 1$. By deleting strings in $V^* \{c\} L_0 \{c\} V^*$ from w we can obtain $\lambda \in (w \Rightarrow M)$, contradicting the relation $x \neq \lambda$. □

4. THE PARALLEL DELETION NUMBER OF A WORD

The *deletion number*, [2], associated to a word w equals the cardinality of the largest deletion set arising from w, that is

$$d(w) = \max \{\text{card}(w \to L) \mid L \subseteq V^*\}.$$

The *parallel deletion number* is defined analogously,

$$\text{pd}(w) = \max \{\text{card}(w \Rightarrow L) \mid L \subseteq V^*\}.$$

Upper bounds for $d(w)$, best possible in the general case, were deduced in [2]. For instance, if card $(V) = s$ and $n \equiv r \pmod{s}$, then

$$\max\{d(w) \mid |w| = n\} = n + 1 + \frac{(s-1)n^2 - sr + r^2}{2s}.$$

It is clear that $d(w) = \text{card}(w \to V^*)$. An analogous result does not hold for parallel deletion because, for every w, $(w \Rightarrow V^*) = \{\lambda\}$.

We now begin our investigation concerning the number $\text{pd}(w)$. For the alphabet with only one element, $\text{pd}(w)$ can be computed, but for the general case the question seems not to be simple at all.

THEOREM 4: *If* $w = a^n$, $n \geq 1$, *then* $pd(w) = n$.

Proof: For $w = a$ we have

$$\text{card}(a \Rightarrow \{\lambda\}) = \text{card}(a \Rightarrow \{a\}) = \text{card}(a \Rightarrow \{\lambda, a\}) = 1.$$

For $w = a^n$, $n \geq 2$, consider

$$L = \{\lambda, a^2, a^3, \ldots, a^n\}.$$

Because we can write $a^n = a\lambda a\lambda \ldots a\lambda a$ we obtain $a^n \in (w \Rightarrow L)$. Moreover, for each a^i, $2 \leq i \leq n$, we have $a^n = a\lambda a\lambda \ldots a\lambda a^i$ which implies $a^{n-i} \in (w \Rightarrow L)$ for all $2 \leq i \leq n$. In conclusion,

$$(w \Rightarrow L) = \{\lambda, a, a^2, \ldots, a^{n-2}, a^n\},$$

that is card $(w \Rightarrow L) = n$. □

The previous proof makes essentially use of the existence of the empty string in L (and the non-existence of a in L). However, if we do not allow λ to be in L then computing card $(w \Rightarrow L)$ is much more difficult. As an illustration of this, let us consider the following particular case: $w = a^n$, $L = \{a^2\}$. The reader can verify that we obtain

$$(a^n \Rightarrow a^2) = \begin{cases} \{\lambda, a^2, a^4, \ldots, a^{2t}\}, & \text{if } n = 6t, \quad t \geq 1, \\ \{a, a^3, \ldots, a^{2t+1}\}, & \text{if } n = 6t+1, \quad t \geq 1, \\ \{\lambda, a^2, a^4, \ldots, a^{2t}\}, & \text{if } n = 6t+2, \quad t \geq 0, \\ \{a, a^3, \ldots, a^{2t+1}\}, & \text{if } n = 6t+3, \quad t \geq 0, \\ \{\lambda, a^2, a^4, \ldots, a^{2t+2}\}, & \text{if } n = 6t+4, \quad t \geq 0, \\ \{a, a^3, \ldots, a^{2t+1}\}, & \text{if } n = 6t+5, \quad t \geq 0. \end{cases}$$

hence

$$\operatorname{card}(a^n \Rightarrow a^2) = \begin{cases} t+1, & \text{if } n = 6t, & t \geq 1, \\ t+1, & \text{if } n = 6t+1, & t \geq 1, \\ t+1, & \text{if } n = 6t+2, & t \geq 0, \\ t+1, & \text{if } n = 6t+3, & t \geq 0, \\ t+2, & \text{if } n = 6t+4, & t \geq 0, \\ t+1, & \text{if } n = 6t+5, & t \geq 0. \end{cases}$$

(we delete a certain number of substrings a^2 from a^n and two consecutive substrings a^2 are either neighbouring or they are separated by one occurrence of a; if a^r is in $(a^n \Rightarrow a^2)$, then also a^{r-2} is in $(a^n \Rightarrow a^2)$ because we can arrange the deleted substrings a^2 in such a way as to delete two more symbols a bounding them.)

In the case of arbitrary alphabets with at least two symbols we obtain the following surprising result.

THEOREM 5: *If* $card\,(V) \geq 2$, *then there is no polynomial* f *such that for every* $w \in V^*$ *we have* $pd\,(w) \leq f\,(|w|)$.

Proof: It suffices to show that, given a polynomial f (in one variable), there are strings w such that $\operatorname{pd}(w) > f(|w|)$.

Take a polynomial f of degree $n \geq 1$ and consider the strings

$$w_{n,m} = (a^m b^m)^n.$$

Moreover, take

$$L_m = \{a^i b^j \mid 1 \leq i, j \leq m-1\}$$

and evaluate the cadinality of $(w_{n,m} \Rightarrow L_m)$.

As each string in L_m contains at least one occurrence of a and one occurrence of b, we can delete from $w_{n,m}$ exactly n strings of L_m, which implies

$$(w_{n,m} \Rightarrow L_m) = \{a^{m-i_1} b^{m-j_1} a^{m-i_2} b^{m-j_2} \ldots a^{m-i_n} b^{m-j_n} \mid 1 \leq i_s, j_s \leq m-1,\ 1 \leq s \leq n\}.$$

Consequently,

$$\operatorname{card}(w_{n,m} \Rightarrow L_m) = (m-1)^{2n}.$$

Clearly, because $2n$ is a constant, for large enough m we have

$$\mathrm{pd}\,(w_{n,m}) \geq (m-1)^{2n} > f\,(2nm) = f\,(|w_{n,m}|),$$

which completes the proof. □

5. THE COLLAPSE SET OF A LANGUAGE

We observed in the previous section that, for every word w, $(w \Rightarrow V^*) = \{\lambda\}$. We can express this by saying that every word *collapses* to the empty word when subjected to parallel deletion with respect to V^*. We speak also of the *collapse set* of V^*. Thus, the collapse set of V^* equals V^*.

In general, we define the *collapse set* of a nonempty language $L \subseteq V^*$ by

$$\mathrm{cs}\,(L) = \{w \in V^* \mid (w \Rightarrow L) = \{\lambda\}\}.$$

This language is always nonempty because it contains each of the shortest words in L.

We give first some examples.

(1) $\mathrm{cs}\,(\{a^n b^n \mid n \geq 1\}) = (ab)^+$,

(2) $\mathrm{cs}\,(\{a, bb\}) = a^*bb\,(a^+bb)^*\,a^* \cup a^+$, (hence $\mathrm{cs}\,(L)$ can be infinite for finite L),

(3) $\mathrm{cs}\,(\{ab\} \cup \{a^n b^m a^p \mid n, m, p \geq 1\}) = \{ab\}$, (hence $\mathrm{cs}\,(L)$ can be finite for infinite L),

(4) $\mathrm{cs}\,(\{ca^n b^n \mid n \geq 1\}) = \{ca^n b^n \mid n \geq 1\}^+$, (hence $\mathrm{cs}(L)$ can be nonlinear for linear L).

Moreover, we have

THEOREM 6: *There is a linear language L such that $cs(L)$ is not context-free.*

Proof: Take

$$L = \{dda^n b^m c^n \mid n, m \geq 1\} \cup \{da^n b^m c^p \mid n, m, p \geq 1, m \geq p\}.$$

Clearly, L is linear. Moreover, we have

$$cs\,(L) \cap d^2 a^+ b^+ c^+ = \{d^2 a^n b^m c^n \mid 1 \leq m < n\}$$

and this is not a context-free language (mark the occurrences of b and use Ogden's lemma).

The equality follows from the next three remarks:

(i) all the strings in $\mathrm{cs}\,(L) \cap d^2a^+b^+c^+$ are of the form $d^2a^nb^mc^n$, $n, m \geq 1$;

(ii) for $m \geq n \geq 1$, we have

$$(d^2a^nb^mc^n \Rightarrow da^nb^mc^n) = \{d\},$$

hence $d^2a^nb^mc^m$ is not in $\mathrm{cs}\,(L) \cap d^2a^+b^+c^+$;

(iii) for $1 \leq m < n$, we have

$$(d^2a^nb^mc^n \Rightarrow L) = (d^2a^nb^mc^n \Rightarrow \{d^2a^nb^mc^n\}) = \{\lambda\}. \quad \square$$

THEOREM 7: *Let $L \subseteq V^*$ be an arbitrary language. Then*

$$cs\,(L) = L^+ - M,$$

where

$$M = (V^*L \cup \{\lambda\})\,(V^+ - V^*LV^*)\,(LV^* \cup \{\lambda\}).$$

Proof: "$\subseteq$" Take $x \in \mathrm{cs}\,(L)$. Clearly, $x \in L^+$. Suppose $x \in M$, hence we can write

$$x = x_1uvwx_2$$

with

$$\begin{aligned} &x_1\,u = \lambda \text{ or } x_1 \in V^*, \qquad u \in L,\\ &v \in V^+, \qquad v \notin V^*LV^*,\\ &wx_2 = \lambda \text{ or } w \in L, \qquad x_2 \in V^*. \end{aligned}$$

As $v \neq \lambda$ and v contains no subword of L, there is a string in $(x \Rightarrow L)$ containing the substring v, which implies $x \notin \mathrm{cs}\,(L)$, a contradiction.

"$\supseteq$" Take $x \in L^+ - M$ and assume $x \notin \mathrm{cs}\,(L)$. Therefore there is $z \neq \lambda$, $z \in (x \Rightarrow L)$. Consequently, we can write $z = z_1z_2z_3$, $z_2 \neq \lambda$, $z_1, z_2 \in V^*$, z_2 containing no substring in L and

$$\begin{aligned} &x = x_1uz_2vx_3,\\ \text{with}\quad &x_1u = \lambda \text{ or } x_1 \in V^*, \qquad u \in L,\\ &z_2 \in V^+, \qquad z_2 \notin V^*LV^*,\\ &vx_3 = \lambda \text{ or } v \in L, \qquad x_3 \in V^*, \end{aligned}$$

such that $z_1 z_2 z_3 \in (x \Rightarrow L)$, $z_1 \in (x_1 \Rightarrow L)$, $z_3 \in (x_3 \Rightarrow L)$. In conclusion, $x \in M$, hence $x \notin L^+ - M$, a contradiction. □

COROLLARY 1: *If L is regular (context-sensitive), then $cs(L)$ is also regular (respectively context-sensitive).*

Proof: Obvious, from the closure properties of the families of regular and context-sensitive languages. □

THEOREM 8: *For $L \subseteq V^*$ we have $cs(L) = V^*$ if and only if $V \cup \{\lambda\} \subseteq L$.*

Proof: In general, $\text{cs}(L) \subseteq V^*$. If $V \subseteq L$, then for every $w \in V^+$ we have $(w \Rightarrow L) = \{\lambda\}$, hence $V^+ \subseteq \text{cs}(L)$. If $\lambda \in L$ then $(\lambda \Rightarrow L) = \{\lambda\}$, too. In conclusion, $\text{cs}(L) = V^*$.

Conversely, if $\text{cs}(L) = V^*$, then $V \cup \{\lambda\} \subseteq \text{cs}(L)$. For $a \in V$ we can have $(a \Rightarrow L) = \{\lambda\}$ only if $a \in L$, therefore $V \subseteq L$. Similarly, $(\lambda \Rightarrow L) = \{\lambda\}$ only if $\lambda \in L$ (if $L \subseteq V^+$, then $(\lambda \Rightarrow L) = \emptyset$). □

6. k-PARALLEL DELETION

Another natural way to define a deletion operation, intermediate between the sequential and the parallel ones, is to remove exactly k strings, for a given k. Namely, for $w \in V^*$, $L \subseteq V^*$, $k \geq 1$, write

$$(w \Rightarrow_k L) = \{u_1 u_2 \dots u_{k+1} \mid u_i \in V^*,\ 1 \leq i \leq k+1,\ w = u_1 v_1 u_2 v_2 \dots u_k v_k u_{k+1},\ \text{for } v_i \in L,\ 1 \leq i \leq k\}$$

Sets of this form will be referred to as *k-deletion sets*; for given $k \geq 1$ we denote by E_k the family of k-deletion sets.

THEOREM 9: *For all $k \geq 1$, $E_k \subset E_{k+1}$, strict inclusion.*

Proof: Take $F \in E_k$, $F = (w \Rightarrow {}_k L)$ and construct

$$w' = (w\#)^k\, w\$,$$
$$L' = (vw_2\#w_1v \mid v \in L,\ w = w_1 v w_2\} \cup \{\$\}.$$

We obtain

$$(w' \Rightarrow_{k+1} L') = F.$$

Indeed, each string in L', excepting \$, contains one symbol #, hence deleting $k+1$ strings means to remove k strings $vw_2\#w_1v$ and \$. When deleting

$vw_2\#w_1v$ from $\ldots\#w_1vw_2\#w_1vw_2\#\ldots$, we get $\ldots\#_1w_1w_2\#\ldots$, hence (between the neighbour #) exactly the result of removing v. The previous erasing removes the symbol # in the left of w_1 and a prefix of w_1, the next erasing removes the symbol # in the right of w_2 and a suffix of w_2. What remains corresponds to the removing of k subwords which belong to L, hence we obtain a string in F. The converse inclusion is clearly true, hence $F \in E_{k+1}$.

Consequently, $E_k \subseteq E_{k+1}$.

This inclusion is proper. In order to prove this, consider the language

$$L_k = \{a_1, a_2, \ldots, a_{k+1}\}, \qquad k \geq 1.$$

We have $L_k = (w \Rightarrow_k L)$ for

$$w = a_1a_2 \ldots a_{k+1},$$
$$L = L_k$$

(removing any k symbols from w we get a one-symbol string, in all possibilities).

Assume $L_k \in E_{k-1}$; let w, L be such that $L_k = (w \Rightarrow_{k-1} L)$.

In order to obtain a symbol a_i, $1 \leq i \leq k+1$, we have to write

$$w = z_1 \ldots z_{n_i} a_i z_{n_i+1} \ldots z_{k-1}, \qquad z_j \in L, \quad 1 \leq j \leq k-1.$$

for some $n_i \geq 0$. Consider writings of w of this form (hence decompositions in $k-1$ strings in L and one symbol a_i) for all i, $1 \leq i \leq k+1$. By changing the subscripts of the specified symbols a_i, we may assume that these distinguished occurrences of $a_1, \ldots, a_{k+1}$ appear in w in the natural order,

$$w = w_1a_1w_2a_2 \ldots w_{k+1}a_{k+1}w_{k+2},$$

for $w_i \in V^*$, $1 \leq i \leq k+2$, V being an alphabet including $\{a_1, \ldots, a_{k+1}\}$.

Therefore, for each a_i, $1 \leq i \leq k+2$, we can decompose $w_1a_1 \ldots w_i$ in $n_i \geq 0$ strings in L and $w_{i+1}a_{i+2} \ldots a_{k+1}w_{k+2}$ in $k-1-n_i$ strings in L.

If $n_i \geq n_{i+1}$, then $n_i + k - 1 - n_{i+1} \geq k-1$. Removing t strings from the n_i strings in the left of a_i and s strings from the $k-1-n_{i+1}$ strings in the right of a_{i+1}, with $t+s = k-1$ (this is possible, because we have at least $k-1$ strings at our disposal), we get a string of the form $y_1a_iw_{i+1}a_{i+1}y_2$, $y_1, y_2 \in V^*$, which must be in L_k, a contradiction.

Consequently, $n_i < n_{i+1}$, $1 \leq i \leq k+1$. As $n_1 \geq 0$, we obtain $n_{k+1} \geq k$.

The set L cannot contain the string λ, otherwise by erasing $k-1$ occurrences of λ we get the string w, a contradiction. Therefore, the string $w_1 a_1 \ldots w_{k+1}$ can be decomposed into $n_{k+1} > k-1$ non-empty strings in L. By removing the first $k-1$ of them, we obtain a string of the form $y a_{k+1} w_{k+2}$, $y \in L$, $y \neq \lambda$. Such a string is not in L_k, a contradiction. Consequently, $L_k \notin E_{k-1}$. □

Remark: The extra symbols in the first part of the proof cannot be avoided. For instance, consider the set

$$F = \{a^i \mid 1 \leq i \leq k+1\}, \qquad k \geq 1.$$

We have $F = (w \Rightarrow_k L)$ for

$$w = a^{2k+1},$$
$$L = \{a, aa\},$$

hence $F \in E_k$.

However, there is no $w \in a^*$, $L \subseteq a^*$ such that $F = (w \Rightarrow_j L)$ for $j > k$.

Indeed, assume that such w, L exist and denote

$$M = \max \{i \mid a^i \in L\},$$
$$m = \min \{i \mid a^i \in L\}.$$

By removing j times a^M we must get the shortest string in F, that is a; by removing j times a^m we get the longest string, a^{k+1}. Therefore

$$|w| = M \cdot j + 1 = m \cdot j + k + 1.$$

Thus $(M-m) \cdot j = k$, which is impossible as $j > k$ and $M - m$ is a natural number.

On the other hand, $F = (w \Rightarrow_{k+j} L)$, $j \geq 1$, for

$$w = a^{k+1} b^{k+j},$$
$$L = \{a^i b \mid 0 \leq i \leq k\},$$

hence using one extra symbol we get $F \in E_{k+j}$ for all $j \geq 1$.

THEOREM 10: *For every finite set F, there is a k such that $F \in E_k$.*

Proof: If card $(F) = 1$, $F = \{x\}$, take $w = x$, $L = \{\lambda\}$, and we have $(w \Rightarrow {}_kL) = F \in E_k$ for all $k \geq 1$.

Assume now

$$F = \{x_1, x_2, \ldots, x_k\}, \qquad k \geq 2,$$

and construct

$$w = x_1 \#_1 x_2 \#_2 \ldots \#_{k-1} x_k,$$
$$L = \{x_i \#_i, \#_i x_{i+1} \mid 1 \leq i \leq k-1\}.$$

We have

$$F = (w \Rightarrow_{k-1} L).$$

Indeed, we have to remove $k-1$ substrings of w; each string of L contains a symbol $\#_i$, hence all of them are removed from w; together with $\#_i$ either x_i or x_{i+1} is removed too, hence what remains is a complete string x_j, $1 \leq j \leq k$. Consequently, $F \in E_{k-1}$.

For

$$m = \max \{|x_i| \mid 1 \leq i \leq k\},$$

we can replace the new symbols $\#_i$ by $ba^{m+i}b$, $1 \leq i \leq k$. As such strings appear only once in w and they identify the strings x_i, x_{i+1} in pairs $x_i ba^{m+i}b$, $ba^{m+i}bx_{i+1}$, we obtain $(w \Rightarrow_{k-1} L) = F$ for the modified w, L too. □

In conclusion, we obtain an infinite hierarchy of families of finite languages, lying in between the deletion sets and the parallel deletion sets,

$$D - \text{sets} = E_1 \subset E_2 \subset \ldots \subset \bigcup_{i \geq 1} E_i = PD - \text{sets} = FIN.$$

Therefore, we can define a *complexity measure* for finite languages, say $Del : FIN \to \mathrm{N}$, by

$$Del\,(F) = \min \{k \mid F \in E_k\}.$$

From the previous theorem, if card $(F) \geq 2$, then $Del\,(F) \leq$ card $(F) - 1$ and $Del\,(F) = 1$ for card $(F) = 1$.

In view of the next theorem, $Del\,(F)$ is computable.

THEOREM 11: *Given a set F and a natural number k, it is decidable whether $F \in E_k$ or not.*

Proof: For given F and k, denote

$$m = \text{card}\,(F),$$
$$l = \max\{|v| \mid v \in F\}.$$

It is enough to show that if F is in E_k, then it can be obtained from a string w whose length is at most $(l+1)(2km+1)$ by k-parallel deletion.

To show this, assume F is obtained from a string w whose length is greater than $(l+1)(2km+1)$ by deleting some language L.

Claim: There is a subword u of w with $|u| = l+1$ such that every word in F can be obtained from w by a deletion in which u is a subword of one of the deleted words in L.

Indeed, if we divide w into blocks of length $l+1$, we get at least $2km+1$ blocks. Choose for each word in F an arbitrary way it can be obtained from w and mark each block that contains either a prefix or a suffix of a deleted L-word. In this way at most $2k$ blocks will be marked for each word in F, which means that altogether at most $2km$ blocks will be marked. Therefore at least one block remains unmarked. This is the looked for u, hence we have the claim. (Note that u has to be either completely deleted or not deleted at all – the latter is impossible because u is longer than any of the words in F.)

Now, we can change w into w' by replacing u by a new symbol $\#$. Simultaneously we add to L all words obtained from words of L by replacing one occurrence of u by $\#$. Let L' be this new set. It is clear that the k-parallel deletion of L' from w' gives F: Every word in F is obtained because we can do the same deletion as above except that when deleting the word that removed the block u we use the word containing $\#$ instead.

No more words are obtained. Any deletion that removes $\#$ from w' can be done also with w and F; any deletion that does not remove $\#$ from w' uses only words of L' not containing $\#$, which means that the same deletion can be done in w, leaving u in the result – a contradiction with the fact that the words of F are shorter than u.

So F can be obtained from a shorter word w'. The shortest word from which F can be obtained has to be at most $(l+1)(2km+1)$ symbols long. Consequently, there are only finitely many strings w to be checked, hence the problem whether $F = (w \Rightarrow_k L)$ or not for some w is decidable (L must be included in the set of subwords of w, hence it is also finite). □

7. FINAL REMARKS

Besides k-parallel deletion, we can define $(\leq k)$-deletion, $(\geq k)$-deletion, and (k, k')-deletion, removing at most k strings, at least k strings, and at least k but at most k' strings, respectively. We leave the study of such cases to the reader.

Another possibility is to define the k-parallel deletion in the following "forced" way: for a string w and a language L, write

$$\begin{aligned}(\omega \Rightarrow_k^f L) = \{u_1 u_2 \dots u_{k+1} \mid & w = u_1 v_1 u_2 v_2 \dots u_k v_k u_{k+1}, \\ & v_i \in L,\ 1 \leq i \leq k, \\ & u_i \notin V^* (L - \{\lambda\}) V^*,\ 1 \leq i \leq k+1\}\end{aligned}$$

(the remaining strings u_i do not contain substrings in $L - \{\lambda\}$).

Denote by E'_k, $k \geq 1$, the families of sets obtained in this way.

For a finite set

$$F = \{x_1, x_2, \dots, x_n\}, \qquad n \geq 2,$$

define

$$\begin{aligned}w &= \#_1 x_1 \#_2 x_2 \dots \#_n x_n \#_{n+1}, \\ L &= \{\#_1 x_1 \dots \#_{i-1} x_{i-1} \#_i | 1 \leq i \leq n\} \\ &\quad \cup \{\#_i x_i \dots x_n \#_{n+1} | 2 \leq i \leq n+1\} \\ &\quad \cup \{\#_i | 1 \leq i \leq n+1\}.\end{aligned}$$

We have $F = (w \Rightarrow_2^f L)$ (no symbol $\#_i$ can remain, hence we must remove a prefix $\#_1 x_1 \dots x_i \#_i$ and a suffix $\#_{i+1} x_{i+1} \dots x_n \#_{n+1}$, hence we obtain the string x_{i+1}). Therefore, $F \in E'_2$. If $F = \{x\}$, then we can put $w = x\#$, $L = \{\#\}$, and we obtain $F \in E'_1$.

In conclusion, there is no hierarchy in this case.

REFERENCES

1. L. KARI, On insertion and deletion in formal languages, *Ph. D. Thesis*, University of Turku, 1991.
2. L. KARI, A. MATEESCU, Gh. PAUN, A. SALOMAA, Deletion sets, *Fundamenta Informaticae*, to appear.
3. A. SALOMAA, *Formal Languages*, Academic Press, London, 1973.