PUC

Series: Monografias em Ciência da Computação
Nº 4/78

CHARACTERIZATIONS FOR THE REGULAR PREFIX CODES AND RELATED FAMILIES

by

Paulo A. S. Veloso

Departamento de Informática

Pontifícia Universidade Católica do Rio de Janeiro
Rua Marquês de São Vivente, 225 – ZC-19
Rio de Janeiro – Brasil

Series: Monografias em Ciência da Computação
Nº 4/78

Series Editor: Michael F. Challis

March, 1978

CHARACTERIZATIONS FOR THE REGULAR PREFIX CODES AND RELATED FAMILIES

by

Paulo A. S. Veloso

*This research has been sponsored in part by FINEP, CNPq and Canadian NRC.

ABSTRACT

Intrinsic characterizations by means of analogues of regular expressions are given for six families of regular languages related to the prefix codes, namely their reversals and their closure under union, the right and left ideals and their complements. First, a characterization for the regular prefix codes is obtained, which is then used to characterize the other families. Characterizations by finite automata are also presented.

KEY WORDS

Prefix code, regular expression, regular language, finite automaton, operations on languages, closure properties.

RESUMO

São obtidas caracterizações intrínsicas por análogos de expressões regulares para seis famílias de linguagens regulares relacionadas aos códigos de prefixos: suas transpostas e seu fecho sob união, os ideais à direita e à esquerda e seus complementos. Primeiramente, dá-se uma caracterização para os códigos de prefixos regulares, a qual é usada para caracterizar as outras famílias. Apresentam-se também caracterizações através de autômatos finitos.

PALAVRAS CHAVES

Código de prefixos, expressão regular, linguagem regular, autômato finito, operações sobre linguagens, propriedades de fechamento.

ACKNOWLEDGEMENTS

Some of the results reported herein were obtained in October, 1977 while the author was visiting the Mathematics Department, University of Western Ontario, Canada. Financial support from the Canadian and Brazilian National Research Councils is gratefully acknowledged. The author is indebted to Professor G. Thierrin for many helpful discussions.

CONTENTS

1 - INTRODUCTION

The aim of this paper is to characterize some families of regular sets related to the prefix codes both intrinsically by analogues of regular expressions and by means of their finite automata. The families considered are the suffix codes, the languages of the multiple-entry finite automata of [GILL-KOU 74] and of the simultaneous-entry finite automata [VELOSO 75, 77], the right ideals and their complements and the right power-bounded languages of [THIERRIN 76].

One of the basic results in the theory of finite automata and regular languages is Kleene's intrinsic characterization of the class of languages recognized by finite automata, which we state using the notation

$$\underline{\text{Reg}} = < \emptyset, \Lambda, (\{\sigma\})_{\sigma\in\Sigma} ; \cup, \cdot, \overset{*}{\cdot} >$$

to mean that the family Reg of regular sets over Σ is the smallest family of languages (over Σ) containing $\emptyset$, $\Lambda = \{\lambda\}$ and $\{\sigma\}$ for each $\sigma\in\Sigma$, and closed under the binary operations of union, concatenation and binary star ($A\overset{*}{\cdot}B = A^*.B$). This introduces the notion of regular expressions and allows the use of "induction on the shape of regular expressions" both in proofs and in definitions.

Other families of regular sets have been given similar characterizations. For instance, the regular noncounting languages have been shown to be exactly the star-free ones, i.e.

$$\underline{\text{RNC}} = < \emptyset, \Lambda, (\{\sigma\})_{\sigma\in\Sigma} ; \cup, \cdot, - >$$

where $-$ is the binary operation of set difference. (see, e.g., [McNAUGHTON-PAPERT 71]) . Also, [THIERRIN 73] characterized the regular left convex and strongly convex languages over Σ as follows

$$\underline{\text{RLCv}} = < \emptyset, \Lambda, (\{\lambda,\sigma\})_{\sigma\in\Sigma} ; \cup, \cdot, \overset{*}{\cdot} >$$

and

$$\underline{\text{RStCv}} = < \emptyset, \Lambda, (\{\sigma\})_{\sigma\in\Sigma} ; \cup, \circ, \overset{*}{\circ} >$$

where $A\circ B = A.B\cup A\cup B$

and $A\overset{*}{\circ}B = A^*\circ B$.

A prefix code is a language containing no proper prefix of any of its words. These languages are important in this context partly because any regular set is a finite union of languages of the form $P.Q^*$ with P and Q regular prefix codes (see, e.g., [THIERRIN 69]).

In the next section we characterize the families RPC and RSC of the regular prefix and suffix codes. Then these results are used to characterize RRI and RLI, the regular right and left ideals, respectively, and their complements: RCP and RCS. Finally we obtain characterizations for URPC, the closure of RPC under union.

Here, all languages considered shall be over a fixed finite alphabet Σ. We assume some familiarity with the basic notions of finite automata and regular languages. In particular, by an fa we mean a connected finite automaton $M = < \Sigma, S, f, s_0, F >$ with state set S, transition function $f: S\times\Sigma \to S$, initial state $s_o \in S$ (from which all states are reachable) and accepting set $F\subset S$. As usual, the transition function is extended to $f: S\times\Sigma^* \to S$. and for each state $s\in S$, we define its accepted set $A(s) = \{w\in\Sigma^*/f(s,w)\in F\}$. (For more details see, e.g. [GINZBURG 68].).

In order to improve readability, the body of the paper contains only the statements of the results, their proofs being outlined in the appendix at the end.

2 - PREFIX AND SUFFIX CODES

A prefix code is a language containing no proper prefix of its words, i.e. $P \in \underline{PC}$ iff $P_\pi = \emptyset$ iff $P^\pi = P$, where, by definition, $L_\pi = L \cap L.\Sigma^+$ and $L^\pi = L - L.\Sigma^+$.

The simplest prefix codes are the empty set $\emptyset$ and the singletons (which are the only ones over a one-letter alphabet).

There are many alternative properties characterizing PC; for instance, the distributivity of left concatenation over intersection: $P.(A \cap B) = (P.A) \cap (P.B)$.

It is easy to see that the class PC of prefix codes is closed under subsets, intersection, concatenation and derivatives; but not under union, complementation or star.

The class RPC of regular prefix codes can be characterized in terms of fa's by the following simple property

(2.1) $L \in \underline{RPC}$ iff no final state of M is reachable from any other final state, by a non-null word, for any $L \in \underline{Reg}$.

In order to generate RPC it would be natural to start with the simple prefix codes: $\emptyset$, $\{\lambda\}$ and $\{\sigma\}$, for $\sigma \in \Sigma$, and close them under some operations.

Since RPC is the image of Reg under $^\pi$, it seems natural to define operations ω on PC so that $(A \omega B)^\pi = A^\pi \; \omega \; B^\pi$, for each $\omega \in \{\cup, ., \overset{*}{.}\}$. For the case of union, the operation prefix-union: $A \vee B = (A - B\Sigma^+) \cup (B - A\Sigma^+)$ will satisfy this requirement, for $P \cup Q \in \underline{PC}$ iff $P \cup Q = P \vee Q$ whenever $P, Q \in \underline{PC}$. However, there does not appear to be any natural way to define an operation to correspond to concatenation, for, with $A = \{a, ab\}$, $(A.A)^\pi = \{aa, aba\}$ depends on A, rather than on $A^\pi = \{a\}$ only.

Another approach is suggested by the fact that one can obtain all the finite prefix codes using concatenation and prefix-union. So, all we should need is an operation giving infinite prefix codes. The definition of arrow: $A \uparrow B = (A^*.B)^\pi$ is quite natural, closure being automatic as $P \overset{*}{.} Q \in \underline{PC}$ iff $P \overset{*}{.} Q = P \uparrow Q$.

To see that we actually get all of RPC, notice that the minimal fa of a nonempty $P \in$ RPC has a single final state p, the only state reachable from p being a "sink" q. Using this fact one can show

$$(2.2)\ P = H(s_o,p) \vee \bigvee_{t\in S'} G(s_o,t).[(\Gamma(t,t) \vee \bigvee_{t'\neq t} \Gamma(t,t').G(t',t)) \uparrow H(t,p)]$$

where $S' = S - \{p,q\}$; and

(a) all $H(s,t)$ and $\Gamma(s,t)$ are finite prefix codes;

(b) each $G(s,t) \in$ RPC and has an fa with fewer states than P, whenever $s \neq t$.

Thus, we have

$$(2.3)\ \underline{RPC} = < \emptyset , \Lambda , (\{\sigma\})_{\sigma\in\Sigma} ; \vee , . , \uparrow >$$

A suffix code is a language L containing no proper suffixes of its words, i.e. $L \cap \Sigma^{+}.L = \emptyset$. So, reversal establishes a bijection between PC and the class SC of suffix codes, inducing the operations $\wedge$, $\dot{\div}$ and $\downarrow$, where $A \dot{\div} B = B.A$, but we may clearly replace $\dot{\div}$ by concatenation. So, calling $L^{\gamma} = \Sigma^{*}.L$ and $L^{\delta} = L - \Sigma^{+}.L$

$$(2.4)\ \underline{RSC} = < \emptyset , \Lambda , (\{\sigma\})_{\sigma\in\Sigma} ; \wedge , . , \downarrow >,$$

where $A \wedge B = (A-\Sigma^{+}.B) \cup (B-\Sigma^{+}.A)$
and $A \downarrow B = (B.A^{*})^{\delta}$

A characterization of RSC in terms of fa's is the following

(2.5) For a nonempty $R \in$ Reg, $R \in$ RSC iff

(a) the initial state s_o lies in no cycle, i.e. $s_o \notin f(S\times\Sigma^{+})$ and

(b) $A(t)$ is disjoint from R, whenever $t \neq s_o$.

3 - RIGHT AND LEFT IDEALS

If we define $L^{\rho} = L.\Sigma^{*}$ then the right ideals are the languages of the form $R = G^{\rho}$, for some generating language G. Some simple properties of this operation $^{\rho}$ are:

(a) $(A^{\rho})^{\rho} = A^{\rho}$
(b) $(A\cup B)^{\rho} = A^{\rho}\cup B^{\rho}$
(c) $(A\cap B)^{\rho} \subset A^{\rho}\cap B^{\rho}$
(d) $(AB)^{\rho} = AB^{\rho}$
(e) $(A^{+})^{\rho} = A^{\rho} = (A^{\rho})^{+}$

So the class RI of right ideals is closed under union, intersection and plus, and $L \in$ RI iff $L = L^{\rho}$.

Property (d) suggests that we might face a difficulty similar to the one encountered in the case of RPC. However, $L^{\pi\rho} = L^{\rho}$ and $L^{\rho\pi} = L^{\pi}$, so every prefix code generates a right ideal, which has a unique generator in PC. Thus, we have a bijection between RPC and RRI inducing the operations union, $A ¢ B = (A^{\pi}.B^{\pi})^{\rho} = A^{\pi}.B$ and $A \$ B = (A^{\pi}\uparrow B^{\pi})^{\rho} = (A^{\pi})^{*}.B$ on RRI. So

(3.1) $\text{RRI} = < \emptyset \; , \; \Sigma^{*}, \; (\{\sigma\}.\Sigma^{*})_{\sigma\in\Sigma} \; ; \; \cup \; , \; ¢ \; , \; \$ >$

Clearly, RRI has a simple automaton-theoretic characterization:

(3.2) For $L \in$ Reg

$L \in$ RRI iff only final states are reachable from final states, in M.

A left ideal is a language of the form $L = \Sigma^{*}.G$ for some generating language G, i.e. $L = \Sigma^{*}.L$. Thus the left ideals are the reversals of the right ideals, whence

(3.3) $\text{RLI} = < \emptyset \; , \; \Sigma^{*} \; , \; (\Sigma^{*}.\{\sigma\})_{\sigma\in\Sigma} \; ; \; \cup \; ; \; \% \; , \; £ >$

where $A \% B = B.A^{\delta}$
and $A £ B = B.(A^{\delta})^{*}$.

The regular left ideals are the languages of the simultaneous- entry finite automata of [VELOSO 75,77], in view of the following characterization

(3.4) For $L \in \underline{Reg}$

$L \in \underline{RLI}$ iff $L = \bigcap_{s \in S} A(s)$

4 - LANGUAGES CLOSED UNDER PREFIX AND SUFFIX

A language L is closed under prefix (resp. suffix) iff whenever $uw \in L$ then $u \in L$ (resp. $w \in L$). Clearly, both classes CP and CS are closed under union, intersection, concatenation and star. Of course, we have bijections between CP and CS, which is reversal, and between CP and RI (resp. CS and LI), namely complementation. Thus

(4.1) $\underline{RCP} = < \Sigma^* , \emptyset , (\hat{\sigma})_{\sigma \in \Sigma} ; \cap , \nabla , ! >$

where $\hat{\sigma} = \overline{\{\sigma\}.\Sigma^*} = \Lambda \cup (\Sigma - \{\sigma\}).\Sigma^*$;

$A \nabla B = \Sigma^* - (\bar{A}^{\pi}.\bar{B}) = A \cup (\bar{A}^{\pi}.B)$;

$A ! B = \Sigma^* - (\bar{A}^{\pi})^*.\bar{B}$.

Similarly

(4.2) $\underline{RCS} = < \Sigma^*, \emptyset, (\check{\sigma})_{\sigma \in \Sigma} ; \cap , \Delta , ; >$

where $\check{\sigma} = \overline{\Sigma^*\{\sigma\}} = \Lambda \cup \Sigma^*.(\Sigma - \{\sigma\})$;

$A \Delta B = \Sigma^* - (\bar{B}.\bar{A}^{\delta}) = A \cup (B.\bar{A}^{\delta})$;

$A ; B = \Sigma^* - \bar{B}.(\bar{A}^{\delta})^*$.

Automaton-theoretic characterizations for these classes are simple.

(4.3) For any $L \in \underline{Reg}$

$L \in \underline{RCP}$ iff no final state is reachable from any non-final state.

and

(4.4) For any $L \in \underline{Reg}$

$L \in \underline{RCS}$ iff $L = \bigcup_{s \in S} A(s)$ ([GILL-KOU 74])

The characterizations (4.1) and (4.2) have some drawbacks: the operations are somewhat involved and it would be nicer to employ union rather than intersection. Perhaps these can be alleviated by a possible characterization of RCP by means of pairs of prefix codes.

5 - RIGHT POWER-BONDED LANGUAGES AND FINITE UNIONS OF PREFIX CODES.

A language L is right power-bounded iff there exists $n>0$ such that whenever $uv^k \in L$ with $v \neq \lambda$ then $k \leq n$. So, these are special right noncounting languages [SHYR-THIERRIN 75]. The name finite union of prefix codes is self-explanatory. Clearly, both classes RPB and UPC are closed under subsets, union, intersection, and concatenation; but not under complementation or star.

[THIERRIN 76] gives an algebraic proof that, for regular sets, both classes coincide (RRPB=URPC). This also follows from our automaton-theoretic characterization.

(5.1) The following are equivalent for $L \in$ Reg

(a) $L \in$ RRPB;

(b) no final state is reachable from itself;

(c) the set of all words sending s_0 to t is in RPC, for each final state t;

(d) $L \in$ URPC.

Let us call U^nPC the class of all unions of $n>0$ prefix codes. Then, it is clear that $U^1PC = PC$. Also, using L_π and L^π as defined in section 2, we can show

(5.2) For any $n>0$

$$L \in U^{(n+1)}PC \quad \text{iff} \quad L_\pi \in U^nPC.$$

Thus, we can get our characterization

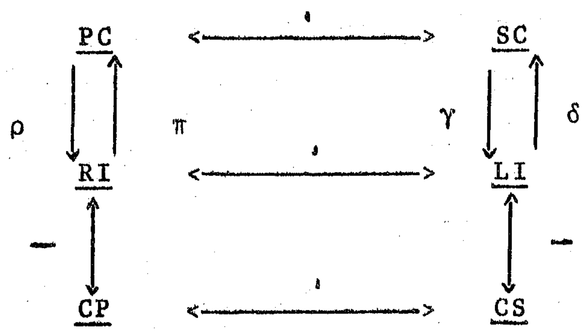
(5.3) RRPB = URPC = $< \emptyset , \Lambda , (\{\sigma\})_{\sigma\in\Sigma} ; \cup , \cdot , ? >$

where $A?B = A^\pi + B$.

Of course, we can introduce the dual concept of left power-bounded language and use reversal to show that the regular ones are exactly the finite unions of regular suffix codes , thereby obtaining their regular-expression characterization similarly to (5.3).

6 - CONCLUSION

The first six families of languages we have considered are related by bijections as displayed below, where ' denotes reversal and $^-$ denotes complementation with respect to Σ^*.



Our approach may be summarized as follows. Once we have our characterization for RPC, we view it as an algebra, then we use the bijections to induce operations on the other families, so as to make the bijections into isomorphisms of the corresponding algebras.

The relation between PC and UPC is slightly different. UPC is connected to finite sequences of prefix codes.

It should be remarked that all these intrinsic characterizations are based upon a corresponding one for RPC. Other characterizations for RPC are possible and they may yield simpler descriptions for some of these families.

Thus, we have characterized six families of regular languages related to the prefix codes, both by means of their fa's and intrinsically by analogues of regular expressions. The latter are not intended to be suited to algebraic manipulations, as this was not the main purpose.

In order to get a better perspective, let us consider as regular expressions over Σ all terms constructed with the constant symbols ϕ, θ and $\underline{\sigma}$, for each $\sigma \in \Sigma$ and the binary operation symbols +, x, †. Several families of regular sets consist exactly of the values of the regular expressions under diverse interpretations, as displayed in the table of valuations below.

	ϕ	θ	$\underline{\sigma}$	+	x	†
<u>Reg</u>	∅	Λ	{σ}	∪	.	$\overset{*}{\cdot}$
<u>RPC</u>	∅	Λ	{σ}	∨	.	↑
<u>RSC</u>	∅	Λ	{σ}	∧	.	↓
<u>RRPB</u>=<u>URPC</u>	∅	Λ	{σ}	∪	.	?
<u>RRI</u>	∅	Σ^*	$\{\sigma\}.\Sigma^*$	∪	¢	$
<u>RLI</u>	∅	Σ^*	$\Sigma^*.\{\sigma\}$	∪	%	£
<u>RCP</u>	Σ^*	∅	$\overline{\{\sigma\}.\Sigma^*}$	∩	∇	!
<u>RCS</u>	Σ^*	∅	$\overline{\Sigma^*.\{\sigma\}}$	∩	Δ	¡
<u>RNC</u>	∅	Λ	{σ}	∪	.	−
<u>RLCv</u>	∅	Λ	{λ,σ}	∪	.	$\overset{*}{\cdot}$
<u>RStCv</u>	∅	Λ	{σ}	∪	o	$\overset{*}{o}$

APPENDIX: Proofs of the results

In this appendix we outline the proofs of our results.

(2.1)

Clear, for $t=f(s_o,u)\in F$ and $f(t,v)=v$ iff $u,uv\in L$.

QED

(2.2)

We associate with each pair of states $s,t\in S'$ the following three prefix codes

- $G(s,t)$, consisting of all non-null words taking s to t for the first time, i.e.
  $G(s,t) = \{w\in\Sigma^+/f(s,w) = t \ \& \ \forall u,v\in\Sigma^+(w=uv \rightarrow f(s,u)\neq t)\}$;
- $H(s,t)$, consisting of all words taking s to t without repeating states, i.e.
  $H(s,t) = \{\sigma_1\ldots\sigma_k\in\Sigma^k/f(s,\sigma_1\ldots\sigma_k) = t \ \& \ \forall\ i<j\leq k\ f(s,\sigma_1\ldots\sigma_i)\neq$
  $\neq f(s,\sigma_1\ldots\sigma_j) \ \& \ k\in N\}$;
- $\Gamma(s,t)$ consisting of all the letters taking s to t, i.e.
  $\Gamma(s,t)=\{\sigma\in\Sigma/f(s,\sigma)=t\}$.

Hence

(a) $H(s,t)$ and $\Gamma(s,t)$ are finite prefix codes, thus expressible with $\emptyset$, Λ, and $\{\sigma\}$, for $\sigma\in\Sigma$, using prefix-union and concatenation.

Furthermore, it is clear that
$G(t,t) = \Gamma(t,t) \cup \bigcup_{t'\neq t} \Gamma(t,t').G(t',t)$.

Now, since all the languagues involved are prefix codes, we may interchange $\cup$ with $\vee$ and $\overset{*}{.}$ with $\uparrow$. So, all that remains to show is

(b) For $s\neq t$, $G(s,t)$ has an fa with fewer states than the minimal one for P.

and

(c) $P = H(s_o,p) \cup \bigcup_{t\in S'} [G(s_o,t).G(t,t)^*.H(t,p)]$

To see (b), transform the fa M for P into $M' = \langle\Sigma, S-\{p\}, g, s, \{t\}\rangle$, where for all $\sigma\in\Sigma$: $g(t,\sigma)=q$ and for $r\neq t$ we put $g(r,\sigma) = f(r,\sigma)$ if $f(r,\sigma)\neq q$ and otherwise $g(r,\sigma)=q$. Then, by induction on w, $g(s,w)=t$ iff $f(s,w)=t$ and for any proper prefix u of w $f(s,u)\neq t$. Thus, M' is a $(|S|-1)$-state fa for $G(s,t)$.

As for (c), consider a word $w=\sigma_1\ldots\sigma_k \in P$ with length $|w|=k>0$ and, for each $j=1,\ldots,k$, put $s_k=f(s_o,\sigma_1\ldots\sigma_j)$. Noting that $s_k=p$ and for all $j<k$ $s_j\in S'$, we have two cases

(i) Either for all $i<j<k$ $s_i\neq s_j$. Then $w\in H(s_o,p)$.

(ii) Or else, for some $i<j<k$ $s_i=s_j$. Then, let r be the maximum among such j's and let $j_o<j_1<\ldots<j_m=r$ be all the i's such that $s_i=t$, where $t=s_r$. Calling $x=\sigma_1\ldots\sigma_{j_o}$ and $z=\sigma_{r+1}\ldots\sigma_k$ we have $x\in G(s_o,t)$ and $z \in H(t,p)$. Now for $i=0,1,\ldots, m-1$, set $y_{i+1}=\sigma_{j_i+1}\ldots\sigma_{j_{i+1}}$; so $y_{i+1} \in G(t,t)$. Thus $w=x.y_1\ldots y_m.z \in$ $\in G(s_o,t).G(t,t)^*.H(t,p)$.

As the other inclusion is clear this completes the proof of claim (c) and of (2.2).

QED

(2.3)

Let E denote the righthand side of (2.3). For each $P\in$<u>RPC</u> let n(P) be the number of states of its minimal fa. We proceed by induction on n(P). If $n(P)=1$, then $P=\emptyset\in E$. Now, let $n(P)=m>1$. Using (2.2), we have for $s,t\in S'$: (a) $H(s,t)$, $\Gamma(s,t)$ are in E; (b) if $s\neq t$ then $n[G(s,t)]<m$ and thus, by induction, $G(s,t)\in E$.

Therefore <u>RPC</u> $\subset$ E, the other inclusion being clear. QED

(2.4)

Follows from (2.3), as reversal establishes an isomorphism between the algebra of (2.3) and that of (2.4) with $\div$ in lieu of concatenation.

QED

(2.5)

(=>) Pick $w \in R = A(s_o)$. Given $t \in S$, we have $u \in \Sigma^*$ such that $t = f(s_o, u)$. Now, to see (a), it suffices to notice that if $s_o = f(t, v)$, for some $v \in \Sigma^+$, then both w, $uvw \in A(s_o)$. As for (b), if $v \in A(t) \cap \cap A(s_o)$ then $uv \in A(s_o)$, so $u = \lambda$ and $t = s_o$.
(<=) If v, $uv \in A(s_o)$ then, with $t = f(s_o, u)$, we have $v \in A(s_o) \cap A(t)$, so by (b), $t = s_o$, whence by (a) $u = \lambda$.

QED

(3.1)

For any $P, Q \in$ PC we have

(a) $(P \vee Q)^\rho = (P - Q.\Sigma^+)\Sigma^* \cup (Q - P.\Sigma^+).\Sigma^* = P^\rho \cup Q^\rho$;

(b) $(P.Q)^\rho = P.Q^\rho = P^{\rho\pi}.Q^\rho = P^\rho ¢ Q^\rho$;

(c) $(P \uparrow Q)^\rho = (P^*.Q)^{\pi\rho} = (P^{\rho\pi})^*.Q^\rho = P^\rho \$ Q^\rho$.

Thus, the assignment $P \mapsto P^\rho$ gives a bijective homomorphism of the algebra of (2.3) onto that of (3.1).

QED

(3.2)

Clear, since $t = f(s_o, u) \in F$ and $f(t, v) \in F$ iff u, $uv \in L$.

QED

(3.3)

Reversal establishes an isomorphism between the algebras of (3.1) and (3.3).

QED

(3.4)

Since every state $s \in S$ is of the form $s = f(s_o, u)$, with $u \in \Sigma^*$ and, for any $v \in \Sigma^*$, $v \in A(s)$ iff $uv \in A(s_o)$, we have $L^\gamma \subset L$ iff for all $u \in \Sigma^*$ $uA(s_o) \subset A(s_o)$ iff for all $s \in S$ $A(s_o) \supset A(s)$.

QED

(4.1)

Similarly to (3.1), for any $A,B \in$ <u>RRI</u>

(a) $\overline{A \cup B} = \bar{A} \cap \bar{B}$

(b) $\overline{A ¢ B} = \Sigma^* - A^\pi . B = \bar{A} \nabla \bar{B}$

(c) $\overline{A \$ B} = \Sigma^* - (A^\pi)^* B = \bar{A} ! \bar{B}$

Thus, complementation gives an isomorphism between the algebras of (3.1) and (4.1).

So, it remains to check that $\Sigma^* - A^\pi . B = \bar{A} \cup A^\pi . \bar{B}$. By means of the distributivity property of A^π, we get

(i) $(\bar{A} \cup A^\pi . \bar{B}) \cup A^\pi . B = \bar{A} \cup A^\pi . (\bar{B} \cup B) = \bar{A} \cup A^{\pi\rho} = \bar{A} \cup A$;

(ii) $A^\pi . B \subset A^\pi . \Sigma^* = A$, so $A^\pi . B \cap \bar{A} = \emptyset$;

(iii) $A^\pi . B \cap A^\pi . \bar{B} = A^\pi . (B \cap \bar{B}) = A^\pi . \emptyset$.

QED

(4.2)

Reversal gives an isomorphism between the algebras of (4.1) and (4.2).

QED

(4.3)

Similar to (3.2).

QED

(4.4)

Similar to (3.4), see [GILL-KOU 74 or VELOSO 77].

QED

(5.1)

(a=>b) If $t \in F$ and $f(t,v)=t$ with $v \in \Sigma^+$, then for some $u \in \Sigma^*$ $t=f(s_o,u)$ and $uv^k \in L$ for all $k \geq 0$.

(b=>c) If $f(s_o,v)= t = f(s_o,vw)$ then $f(t,w)=t$.

(c=>d) Clear.

(d=>a) To show <u>RPC</u> $\subset$ <u>RRPB</u>. We use a "pumping lemma" argument on the fa M for $P \in$ <u>RPC</u>.

Given $u, v \in \Sigma^*$, call $t_k = f(s_o, uv^k)$, for $k=0,1,2,\ldots$. Now, if M has n states and $uv^k \in P$ with $k \geq n$ then for some $0 \leq i < j \leq k$, $t_i = t_j$; so $v = \lambda$.

QED

(5.2)

First, notice that $L = L^\pi \cup L_\pi$ with $L^\pi \in$ PC. So

(a) If $L_\pi \in U^n PC$ then $L \in U^{(n+1)} PC$

Now, to see the converse it suffices to show

(b) If $L \notin U^n PC$ then there exist $w_o, w_1, \ldots, w_n \in \Sigma^+$ such that $w_o, w_o w_1, \ldots, w_o w_1 \ldots w_n \in L$

For $n=1$, it is clear. As for $n>1$, if $L \notin U^n PC$ then by (a) $L_\pi \notin U^{(n-1)} PC$, so by induction we have $w_1, w_1 w_2, \ldots, w_1 w_2 \ldots w_n \in L_\pi$; but then, for some $u \in L$ and $v \in \Sigma^+$, $w_1 = uv$ and $u, uv = w_1, w_1 w_2, \ldots, w_1 w_2 \ldots w_n$ are all in L.

QED

(5.3)

Call G the righthand side of (5.3) and let F be the algebra obtained by replacing prefix-union by set-theoretical union in the righthand side of (2.3). Then, the proof of (2.3), namely claim (c), shows RPC $\subset$ F, so RPC $\subset$ G. Now, by induction using (5.2), we have URPC $\subset$ G. The other inclusion is clear, since for all $A, B \in G$ $A?B \in$ RPC.

QED

REFERENCES

- GILL, A & KOU, L.T. Multiple-entry finite automata. JCSS. New York 9(1): 1-19, Aug. 1974.

- GINZBURG, A. Algebraic theory of automata. New York, Academic Press, 1968.

- McNAUGHTON, R & PAPERT, S. Counter-free automata. Cambridge, Mass., the MIT Press, 1971.

- SHYR, H.J. & THIERRIN, G. Left-noncounting languages. Intern J. of Computer and Inform. Sci. New York 4(1): 95-102, Mar. 1975.

- THIERRIN, G. Decompositions des langages réguliers. R.I.R.O. Paris 3e année (R3): 45-50, Oct. 1969.

- THIERRIN, G. Convex languages. In: NIVAT, M. ed. Automata, languages and programming. Amsterdam, North-Holland, 1973, p.481-492.

- THIERRIN, G. Regular prefix codes and right power-bounded languages. Semigroup Forum. New York 13(1): 77-83, 1976.

- VELOSO, P. A. S. Networks of finite-state machines. Berkeley, University of California, Department of Eletrical Engineering and Computer Sciences, 1975. Ph.D. dissertation.

- VELOSO; P. A. S. Some properties of multiple-entry and simultaneous-entry finite automata. Rio de Janeiro. Universidade Federal do Rio de Janeiro, COPPE, 1977. PTC 2-77.